Naval Research Laboratory

Monterey, CA 93943-5502

NRL/MR/7532/97/7230

A Technical Description of the NRL Adjoint Modeling System

THOMAS E. ROSMOND

Atmospheric Dynamics & Prediction Branch
Marine Meteorology Division

April 1997

19971027 064

Approved for public release; distribution unlimited.

DTIC QUALITY INSPECTED 3

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. Agency Use Only *(Leave Blank)*.	2. Report Date. April 1997	3. Report Type and Dates Covered. Final

4. Title and Subtitle. A Technical Description of the NRL Adjoint Modeling System	5. Funding Numbers. PE 0602435N PN 035-71
6. Author(s). Thomas E. Rosmond	
7. Performing Organization Name(s) and Address(es). Naval Research Laboratory Marine Meteorology Division Monterey, CA 93943-5502	8. Performing Organization Reporting Number. NRL/MR/7532- - 97- - 7230
8. Sponsoring/Monitoring Agency Name(s) and Address(es). Office of Naval Research (ONR)	10. Sponsoring/Monitoring Agency Report Number.
11. Supplementary Notes.	
12a. Distribution /Availability Statement. Approved for public release; distribution unlimited	12b. Distribution Code.

13. Abstract *(Maximum 200 words)*.

This report describes the tangent linear model (TLM), and the accompanying adjoint model, for the Navy Operational Global Atmospheric Prediction System (NOGAPS) spectral forecast model. Details are given for the linearization of the model equations, the development of the matrix transpose operators that constitute the adjoint model, and a discussion of the self-adjoint properties of the spherical harmonic transforms, unique to spectral models such as NOGAPS. A powerful application of the TLM and adjoint models, the calculation of singular vectors about a basic state trajectory forecast of the non-linear NOGAPS model, is also described. Some illustrative examples of singular vector and adjoint model calculations are given . Full model codes are available by request from the author .

14. Subject Terms. A Technical Description of the NRL Adjoint Modeling System	15. Number of Pages. 62
	16. Price Code.

17. Security Classification of Report. UNCLASSIFIED	18. Security Classification of This Page. UNCLASSIFIED	19. Security Classification of Abstract. UNCLASSIFIED	20. Limitation of abstract. Same as report

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

Contents

1 Introduction **1**

2 The NOGAPS spectral model **3**

2.1 *The basic equations* 4

2.2 *Vertical differencing* 7

2.3 *Spectral transforms* 11

2.4 *Time integration* 14

3 The tangent linear model **15**

3.1 *Spectral to real transforms* 17

3.2 *Terrain pressure dependent linearizations* 17

3.3 *The hydrostatic equation* 19

3.4 *The potential temperature and moisture equations* 19

3.5 *The vorticity and divergence equations* 20

3.6 *Diabatic forcing terms* 21

3.7 *Real to spectral transforms and time stepping* 24

4 The adjoint model **24**

4.1 *Time step and time filter adjoint* 25

4.2 *Diabatic forcing adjoint* 28

4.3 *Vorticity and divergence adjoint equations* 29

4.4 *Potential temperature adjoint equations* 32

4.5 *Hydrostatic equation adjoint* 33

4.6 *Vertical velocity and terrain pressure adjoints* 34

4.7 *Exner function adjoints* 35

4.8 *Spectral transform adjoints* 36

5 **Testing of TLM and adjoint models** **38**

6 **The singular vector system** **41**

6.1 *Energy norm formulation* 42

6.2 *Singular vector formulation* 43

6.3 *The Lanczos algorithm* 43

6.4 *The local projection operator* 45

7 **Example Calculations** **46**

7.1 *Singular vectors* 46

7.2 *Adjoint sensitivities* 49

8 **Summary** **50**

1 Introduction

Understanding and predicting the growth of unstable disturbances in the Earth's atmosphere is recognized by meteorologists as perhaps the most important factor in successful weather prediction. The pioneering work on baroclinic instability by Charney(1947) and Eady(1949) demonstrated that the eigenfunctions of simple linearized forms of the equations of motion yield eigenvalues which correspond surprisingly well with observed time and space scales of mid-latitude disturbances. In subsequent years linear stability analysis of various forms of the linearized equations became the foundation for rapid advances in the understanding of atmospheric instability. Beginning in the 1950's numerical computation made a critical contribution to this progress by making it possible to solve eigenvalue/eigenvector problems that were intractable with analytic solution methods.

All of the early studies of atmospheric instability were based on finding the normal mode eigenfunctions of some form of a linear evolution operator $\mathbf{A}$, formulated for a steady state mean flow. However, Lorenz (1965) showed that the eigenvalues of $\mathbf{A^T A}$, called the singular value eigenvalues of $\mathbf{A}$, can be much larger than the normal mode eigenvalues of $\mathbf{A}$ itself. $\mathbf{A^T}$ is the transpose, or adjoint, of $\mathbf{A}$. The eigenvectors associated with the singular values are the singular vectors of $\mathbf{A}$. In a meteorological sense the largest singular values and their associated singular vectors are optimal in that they maximize linear disturbance growth over a given evolution (prediction) period. Also, because the product operator $\mathbf{A^T A}$ is symmetric, solutions for non-steady basic states are potentially more tractable, at least in a linear algebra sense. However, the completely general singular vector problem, for an $\mathbf{A}$ with a realistic time and space varying basic state trajectory, produces linear systems of enormous size that require considerable computational effort and special methods for solution. For example, the evolution operator $\mathbf{A}$ for a numerical weather prediction model of relatively modest resolution can easily yield linear systems of $\mathbf{O(10^7)}$. Only recently,

therefore, has computational power made practical meteorological applications of singular vectors feasible. Lacarra and Talagrand (1988) made singular vector calculations for a barotropic shallow water model, and Molteni and Palmer (1993) did similar calculations with a barotropic model and a 3-level quasi-geostrophic model. Buizza *et al.*(1993) were the first to compute singular vectors for a low resolution (T21) numerical weather prediction (NWP) model based on the primitive equations. Ehrendorfer and Errico (1994) made similar calculations for a limited area NWP model.

Singular vectors have potential important application to a wide variety of practical meteorological problems. In recent years operational NWP centers have begun running ensemble prediction systems, in which each member of the ensemble is an NWP model integration run from initial conditions which differ from those of other ensemble members by some suitably chosen perturbation vector. Ideally, the envelope of forecast trajectories from the ensemble forecasts will occupy the full uncertainty spectrum for this set of numerical forecasts. Singular vectors are a particularly attractive source of these perturbations because they are orthogonal to one another and represent the potentially fastest growing modes of instability for their associated basic states. These instabilities can be expected to give maximum divergence of forecasts running from initial conditions which differ by combinations of singular vectors.

Singular vectors are also important tools for analysis of numerical model prediction error. It is commonly understood that uncertainty in the initial conditions (analysis error) contribute to the forecast errors that grow with time in an NWP model. However, knowledge of the geographical distribution of analysis error is not sufficient for understanding of why these forecast errors grow. It is also necessary to know where the model is sensitive to errors in its initial conditions. The geographical co-location of analysis error and forecast model error sensitivity is the critical combination that leads to rapid growth of forecast error. Singular vectors associated with the largest singular values are ideal representations of the

forecast model error sensitivity patterns, since each singular vector shows the geographical distribution of perturbation growth for the associated singular value.

This report describes the formulation of the singular vector system based on the Navy Operational Global Atmospheric Prediction System (NOGAPS) spectral forecast model (Hogan and Rosmond 1991). In section 2 a brief overview of the NOGAPS spectral model is given. Section 3 describes the linearized version of the NOGAPS spectral model about a time and space varying basic state, often called the tangent-linear model (TLM). Section 4 describes the adjoint model, which is the implicit matrix transpose of the TLM. Section 5 describes the procedures used for testing TLM and adjoint models and presents some testing results. Section 6 describes the formulation of the singular vector system, which is a linear operator based on a combination of the TLM and adjoint operators. Section 7 presents the results of example singular vector calculations. Section 8 is a summary and gives potential future directions for singular vector and adjoint model applications.

2 The NOGAPS spectral model

The basis of the singular vector system is the non-linear model from which the TLM and adjoint models are developed. We use the NOGAPS spectral forecast model, described in detail by Hogan *et al.* (1991). The model is representative of the sophisticated global atmospheric models run by major operational weather forecast centers and meteorological research groups around the world for numerical weather prediction, data assimilation, and climate simulation. NOGAPS is the heart of the Navy's operational weather prediction system run at Fleet Numerical Meteorological and Oceanographic Center (FNMOC). The model contains diabatic parameterizations of all of the important physical processes of the atmosphere, such as cumulus parameterization, cloud and radiation interactions, and the planetary boundary layer. Linearizations of all of these parameterizations would be desir-

able, but the complexity of these schemes is daunting, and only a few attempts at linearizing diabatic processes for NWP models have been made, e.g. Vukićević and Errico (1993). Furthermore, mid-latitude atmospheric instabilities are dominated by the dry dynamics, and experience to date (Buizza 1994) has shown that a linear model with no diabatic processes except a very simple planetary boundary layer gives quite satisfactory results.

For the above reasons the linearizations described in the next section will be restricted to the dry dynamics with a simplified planetary boundary layer (PBL), so in this section only the details of the dynamical equations are presented, as formulated for the NOGAPS global spectral model. Complete description of the rest of the model formulation is in Hogan *et al.*(1991). Parts of the model which are already inherently linear, such as the semi-implicit scheme, will not be discussed. Likewise the normal mode initialization, whose non-linear operations are essentially identical to those in the dry dynamics, will not be discussed. Note that even though the linearization is restricted to the dry dynamics, the basic state trajectory forecast about which TLM or adjoint model perturbations vary is always generated by a full physics version of the non-linear model, usually at relatively high resolution. This ensures that the basic state trajectory will give the best possible synoptic forecasts and realistic linear instability growth behavior.

2.1 *The basic equations*

The NOGAPS spectral forecast model is a primitive equation model formulated in spherical coordinates in the horizontal and a hybrid vertical coordinate similar to that described by Simmons and Strüfing (1981). The horizontal coordinates are the longitude, λ, and the latitude, φ. The vertical coordinate is normalized pressure represented by the variable η which ranges from 0.0 at the model top to 1.0 at the surface. If p_{top} is the pressure at the top of the model atmosphere and p_s is the terrain pressure then the pressure p is a function

of η given by

$$p = A(\eta) + B(\eta)\pi, \tag{1}$$

where $\pi = p_s - p_{top}$. The functions $A(\eta)$ and $B(\eta)$ are any two functions defined on the interval 0.0 to 1.0 with the boundary conditions:

$$\left.\begin{aligned} A(0) &= p_{top} \\ A(1) &= p_{top} \\ B(0) &= 0.0 \\ B(1) &= 1.0 \end{aligned}\right\}, \tag{2}$$

where p_{top} is the model's top pressure, p_S the surface pressure and $\pi = p_S - p_{top}$.

The continuity equation for the conservation of mass in this coordinate system is

$$\frac{\partial}{\partial t}\left(\frac{\partial p}{\partial \eta}\right) + \nabla\cdot\left(\mathbf{u}\frac{\partial p}{\partial \eta}\right) + \frac{\partial}{\partial \eta}\left(\dot{\eta}\frac{\partial p}{\partial \eta}\right) = 0, \tag{3}$$

where $\mathbf{u}$ is the horizontal velocity vector. Integrating (3) with the top and bottom boundary conditions

$$\dot{\eta}(0) = \dot{\eta}(1) = 0, \tag{4}$$

yields the π tendency equation

$$\frac{\partial \pi}{\partial t} = -\int_0^1 \nabla\cdot\left(\mathbf{u}\frac{\partial p}{\partial \eta}\right) d\eta = -\int_{p_{top}}^{p_S} \nabla\cdot(\mathbf{u}dp), \tag{5}$$

where dp a function of λ, φ, and η. We obtain the vertical motion equation by integrating (3) from 0 to η and substituting for the $\partial\pi/\partial t$ term with the right hand side of (5), yielding

$$\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right](\eta) = B(\eta)\int_0^1 \nabla\cdot\left(\mathbf{u}\frac{\partial p}{\partial \eta}\right) d\eta - \int_0^\eta \nabla\cdot\left(\mathbf{u}\frac{\partial p}{\partial \eta}\right) d\eta. \tag{6}$$

The thermodynamic energy equation in terms of potential temperature θ is

$$\frac{\partial \theta}{\partial t} = -\frac{u}{a\cos\varphi}\frac{\partial \theta}{\partial \lambda} - \frac{v}{a}\frac{\partial \theta}{\partial \varphi} - \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial \theta}{\partial p} + Q_\theta, \tag{7}$$

where a is the Earth's radius, u and v are the zonal and meridional wind components, respectively, and Q_θ is the diabatic forcing due to radiation, latent heat release processes, horizontal diffusion, and vertical mixing.

The moisture conservation equation is

$$\frac{\partial q}{\partial t} = -\frac{u}{a\cos\varphi}\frac{\partial q}{\partial \lambda} - \frac{v}{a}\frac{\partial q}{\partial \varphi} - \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial q}{\partial p} + Q_q, \tag{8}$$

where the forcing term Q_q is due to condensation/evaporation processes and turbulent and cumulus vertical mixing.

We write the hydrostatic equation in the form:

$$\frac{\partial \phi}{\partial P} = -c_p\theta, \tag{9}$$

where ϕ is the geopotential and P is the Exner function

$$P = \left(\frac{p}{p_0}\right)^\kappa. \tag{10}$$

In (10) p_0 is 1000 mb and $\kappa = R/c_p$ is the ratio of the gas constant R to the heat capacity c_p.

The dependent variables describing the motion field in NOGAPS are the vorticity ζ and divergence D. This choice is a routine one for global spectral models because as scalar quantities ζ and D are easily expandable in terms of spherical harmonics. To represent the relationship between ζ, D and the horizontal wind components it is convenient to define the operator $\alpha(g,h)$, which operates on any two functions g and h, as

$$\alpha(g,h) = \frac{1}{1-\mu^2}\frac{\partial g}{\partial \lambda} + \frac{\partial h}{\partial \mu}, \tag{11}$$

where $\mu = \sin\varphi$. The vorticity and the divergence are expressed as

$$\zeta = \alpha(V, -U), \tag{12}$$

and

$$D = \alpha(U, V), \tag{13}$$

where we have defined the scaled wind velocity components

$$U = u\frac{\cos\varphi}{a}, \tag{14}$$

and

$$V = v\frac{\cos\varphi}{a}. \tag{15}$$

Similarly, the tendency equations for vorticity and divergence are

$$\frac{\partial\zeta}{\partial t} = -\alpha(G, H), \tag{16}$$

and

$$\frac{\partial D}{\partial t} = \alpha(H, -G) - \nabla^2(\phi + I), \tag{17}$$

where the functions G, H, and I are:

$$G = U(\zeta + f) + \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\left(\frac{\partial V}{\partial p}\right) + (1-\mu^2)\frac{c_p}{a^2}\theta\left(\frac{\partial P}{\partial \pi}\right)\left(\frac{\partial \pi}{\partial \mu}\right) - Q_v\frac{\cos\varphi}{a}, \tag{18}$$

$$H = V(\zeta + f) - \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\left(\frac{\partial U}{\partial p}\right) - \frac{c_p}{a^2}\theta\left(\frac{\partial P}{\partial \pi}\right)\left(\frac{\partial \pi}{\partial \lambda}\right) + Q_u\frac{\cos\varphi}{a}, \tag{19}$$

$$I = \frac{a^2}{1-\mu^2}\left(\frac{U^2+V^2}{2}\right), \tag{20}$$

and f is the coriolis parameter and Q_u and Q_v are diabatic forcing terms due to surface friction and vertical mixing of momentum.

2.2 *Vertical differencing*

NOGAPS uses second order finite differencing for representing vertical derivatives. The distribution of variables on the model's vertical grid is shown in Figure 1. The dashed lines denote 'full' levels and the solid lines 'half' levels which are at the interfaces between model layers. The discrete analogue to (1) is

$$p_{k+1/2} = A_{k+1/2} + B_{k+1/2}\pi, \tag{21}$$

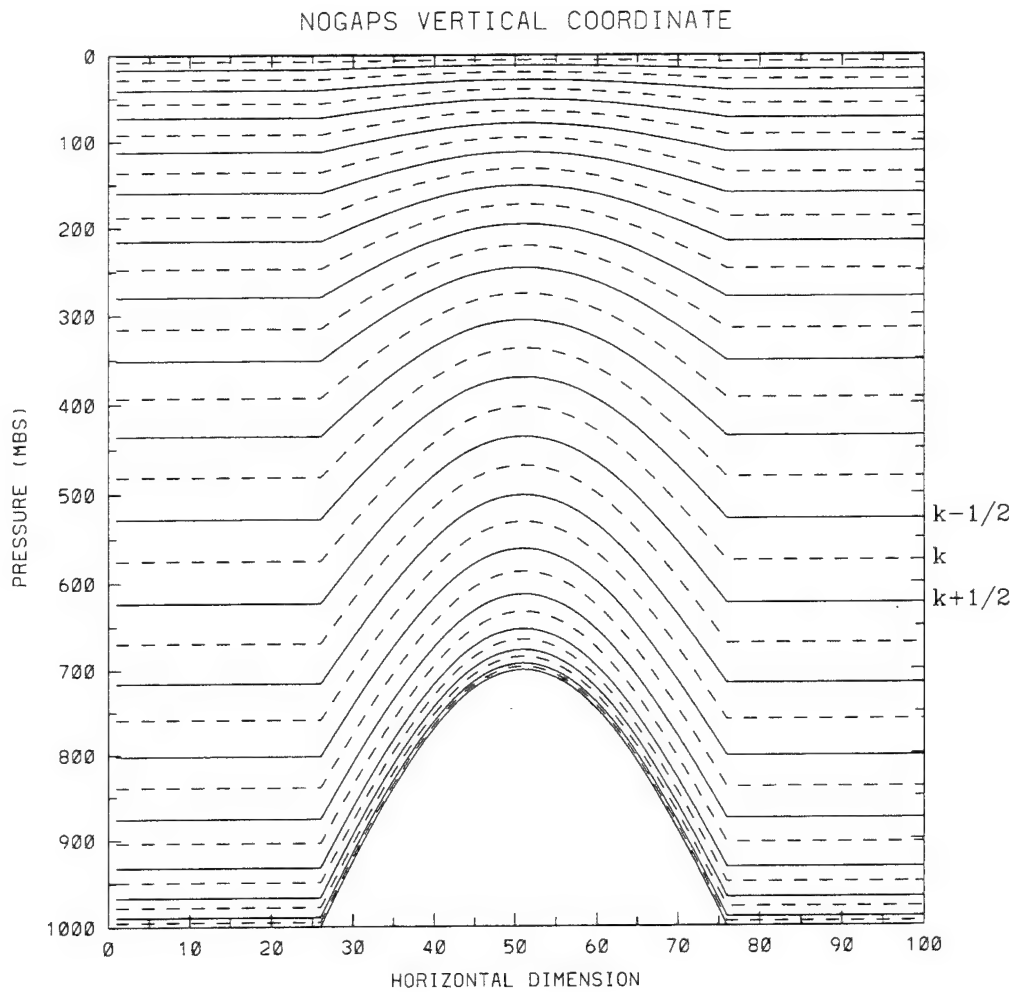

Figure 1: Terrain following vertical coordinate for 18 level NOGAPS model. Solid lines are 'half' levels ,dashed lines are 'full' levels.

subject to the boundary conditions of (2). Model layer thicknesses are given by:

$$\Delta p_k = \Delta A_k + \Delta B_k \pi, \tag{22}$$

where

$$\Delta A_k = A_{k+1/2} - A_{k-1/2},$$

$$\Delta B_k = B_{k+1/2} - B_{k-1/2}.$$

The discrete analogue to (5) is

$$\frac{\partial \pi}{\partial t} = -\sum_{l=1}^{L} \left[\Delta A_l \nabla \cdot \mathbf{u_l} + \Delta B_l \nabla \cdot (\pi \mathbf{u_l})\right], \tag{23}$$

where L is the number of model layers. Analogous to (6) in the continuous case, the vertical velocity in the hybrid coordinate system is

$$\begin{aligned} \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2} &= B_{k+1/2} \sum_{l=1}^{L} \left(\Delta A_l \nabla \cdot \mathbf{u_l} + \Delta B_l \nabla \cdot (\pi \mathbf{u_l})\right) \\ &- \sum_{l=1}^{k} \left(\Delta A_l \nabla \cdot \mathbf{u_l} + \Delta B_l \nabla \cdot (\pi \mathbf{u_l})\right) \end{aligned} \tag{24}$$

At the half pressure level, the Exner function P is:

$$P_{k+1/2} = \left(\frac{p_{k+1/2}}{p_0}\right)^{\kappa}, \tag{25}$$

and at full levels is given by Phillips (1974) as

$$P_k = \frac{1}{(\kappa+1)p_0^{\kappa}} \left(\frac{p_{k+1/2}^{\kappa+1} - p_{k-1/2}^{\kappa+1}}{\Delta p_k}\right). \tag{26}$$

From (21), (25), and (26) the term $(\partial P/\partial \pi)_k$ in (18) and (19) becomes, after a little algebra,

$$\frac{\partial P_k}{\partial \pi} = \left[\frac{B_{k+1/2}\left(P_{k+1/2} - P_k\right) + B_{k-1/2}\left(P_k - P_{k-1/2}\right)}{\Delta p_k}\right]. \tag{27}$$

Vertical advection requires that we define half level values of U, V, θ and q. These choices are important for energy conservation. The general form for vertical advection in the discrete equations is

$$\begin{aligned} \left(\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial X}{\partial p}\right)_k &= \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2} \left(\frac{X_{k+1/2} - X_k}{\Delta p_k}\right) \\ &+ \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2} \left(\frac{X_k - X_{k-1/2}}{\Delta p_k}\right), \end{aligned} \tag{28}$$

where X represents any dynamic variable.

For the thermodynamic energy equation the correct choice for half level θ is

$$\theta_{k+1/2} = \theta_{k+1} + \mathcal{W}_{k+1/2}\left(\theta_k - \theta_{k+1}\right), \tag{29}$$

where

$$\mathcal{W}_{k+1/2} = \frac{P_{k+1/2} - P_k}{P_{k+1} - P_k}. \tag{30}$$

(Haltiner and Williams 1980). This ensures total energy conservation in conversions between kinetic and potential energy in the model atmosphere. Note that this is **not** a linear interpolation in P. Details can be found in Hogan *et al.*(1991). Using (29) the discrete form of (7) is

$$\begin{aligned} \frac{\partial \theta_k}{\partial t} &= \theta_k D_k - \alpha\left(U_k\theta_k, V_k\theta_k\right) \\ &- \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2}\left(\frac{\theta_{k+1/2} - \theta_k}{\Delta p_k}\right) - \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2}\left(\frac{\theta_k - \theta_{k-1/2}}{\Delta p_k}\right) + Q_{\theta_k}, \end{aligned} \tag{31}$$

where we have used the identity $\mathbf{u}\cdot\nabla\theta = \nabla\cdot(\mathbf{u}\theta) - \theta\nabla\cdot\mathbf{u}$ and a generalization of (13) to redefine the horizontal temperature advection terms.

For half level U and V the linear interpolation

$$X_{k+1/2} = \left(\frac{X_k + X_{k+1}}{2}\right), \tag{32}$$

where X is one of the two scaled wind components, guarantees kinetic energy conservation for vertical advection of momentum. This is equivalent to $\mathcal{W} = 0.5$ in (30) . The same form is used to define half level q, although this can lead to problems because it cannot guarantee that positive definite moisture is preserved by the vertical advection. However, because we are mainly concerned with the linearization of the dry dynamics of NOGAPS, we will not concern ourselves with this problem here. Using (32) for defining half level U, V, and q, the discrete forms of (16), (17), (18), (19), (20), and (8) are, respectively,

$$\frac{\partial \zeta_k}{\partial t} = -\alpha\left(G_k, H_k\right), \tag{33}$$

$$\frac{\partial D_k}{\partial t} = \alpha\left(H_k, -G_k\right) - \nabla^2\left(\phi_k + I_k\right). \tag{34}$$

$$G_k = U_k(\zeta_k + f) + \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2}\left(\frac{V_{k+1} - V_k}{2\Delta p_k}\right) + \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2}\left(\frac{V_k - V_{k-1}}{2\Delta p_k}\right)$$

$$
\begin{aligned}
&+ \quad (1-\mu^2)\frac{c_p}{a^2}\theta_k\left(\frac{\partial P}{\partial \pi}\right)_k \frac{\partial \pi}{\partial \mu} - Q_{V_k}\frac{\cos\varphi}{a}, && (35)\\
H_k &= V_k(\zeta_k + f) - \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2}\left(\frac{U_{k+1}-U_k}{2\Delta p_k}\right) - \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2}\left(\frac{U_k - U_{k-1}}{2\Delta p_k}\right)\\
&- \quad \frac{c_p}{a^2}\theta_k\left(\frac{\partial P}{\partial \pi}\right)_k \frac{\partial \pi}{\partial \lambda} + Q_{U_k}\frac{\cos\varphi}{a}, && (36)\\
I_k &= \frac{a^2}{1-\mu^2}\left(\frac{U_k^2 + V_k^2}{2}\right), && (37)
\end{aligned}
$$

and

$$
\begin{aligned}
\frac{\partial q_k}{\partial t} &= q_k D_k - \alpha\left(U_k q_k, V_k q_k\right)\\
&- \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2}\left(\frac{q_{k+1/2} - q_k}{\Delta p_k}\right) - \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2}\left(\frac{q_k - q_{k-1/2}}{\Delta p_k}\right) + Q_{q_k}, && (38)
\end{aligned}
$$

The discrete form of the hydrostatic equation (9) is

$$
\left.\begin{aligned}
\phi_k - \phi_S &= c_p\theta_k(P_{k+1/2} - P_k), \quad k = L\\
\phi_k - \phi_{k+1} &= c_p\theta_{k+1/2}(P_{k+1} - P_k), \quad L-1 \geq k \geq 1
\end{aligned}\right\}, \qquad (39)
$$

where ϕ_S is the terrain geopotential and $\theta_{k+1/2}$ is the half level potential temperature given by (29).

2.3 *Spectral transforms*

The NOGAPS model equations presented above are solved in a sequence of steps which require that model variables be represented both in 'spectral space' and 'grid point space'. Spherical harmonic transforms are the natural choice for the equations expressed in spherical coordinates. Spherical harmonics are the associated Legendre polynomials, $P_n^m(\mu)$ multiplied by the complex Fourier series, $e^{im\lambda}$. The subscript n is the total wavenumber and superscript m the zonal wavenumber. The series are truncated assuming a triangular truncation with M indicating the total number of resolvable waves. If $S_n^m(t)$ is a set of complex spherical harmonic coefficients for a variable X, then the spectral to grid spectral

expansion of X is

$$X(\lambda_l, \mu_j, \eta_k, t) = \sum_{m=-M}^{M} \left[\sum_{n=|m|}^{M} S_n^m(\eta_k, t) P_n^m(\mu_j) \right] e^{im\lambda_l}, \tag{40}$$

where X is any of the model dependent variables π, ζ, D, θ, or q at any time t. The subscripts l and j represent the discrete grid longitudes and latitudes, respectively, of a suitably chosen transform grid, and the η_k are discrete vertical coordinate values. The inner sum in (40) yields a set of complex Fourier coefficients, so the outer summation is a discrete backward Fourier transform which is solved by fast Fourier transform (FFT). The choice of the transform grid resolution associated with a particular value of M is an important one to ensure accuracy and integral property conservation. Details of this choice are in Hogan *et al.*(1991).

The grid point to spectral transform is obtained from (40) by applying the orthogonality property of the spherical harmonics:

$$S_n^m(\eta, t) = \frac{1}{2\pi} \int_{-1}^{1} P_n^m(\mu) \left[\int_0^{2\pi} X(\lambda, \mu, \eta, t) e^{-im\lambda} d\lambda \right] d\mu. \tag{41}$$

The expression inside the square brackets is a forward Fourier transform also solved by FFT. After the FFT, (41) is

$$S_n^m(\eta, t) = \int_{-1}^{1} \mathcal{F}^m[X(\mu, \eta, t)] P_n^m(\mu) d\mu, \tag{42}$$

Where $\mathcal{F}^m[X(\mu, \eta, t)]$ are the coefficients from the Fourier transform described above. The integration in (42) is performed using the method of Gaussian quadrature, for which, given any polynomial $g(\mu)$, of degree $2J-1$ or less, the integral of g from -1 to 1 is computed exactly as

$$\int_{-1}^{1} g(\mu) d\mu = \sum_{j=1}^{J} w_j g(\mu_j),$$

where μ_j are the so-called Gaussian latitudes and w_j the Gaussian quadrature weights. With this the discrete form of (42) is

$$S_n^m(\eta_k, t) = \sum_{j=1}^{(3M+1)/2} w_j \mathcal{F}^m[X(\mu_j, \eta_k, t)] P_n^m(\mu_j). \tag{43}$$

The upper limit on the summation is the number of Gaussian latitudes necessary to ensure that the integral in (42) is evaluated exactly.

Horizontal derivatives of spectral variables are found by differentiating (40), which yields

$$\frac{\partial X}{\partial \lambda}(\lambda_l, \mu_j, \eta_k, t) = \sum_{m=-M}^{M} im \left[\sum_{n=|m|}^{M} S_n^m(\eta_k, t) P_n^m(\mu_j) \right] e^{im\lambda_l}, \tag{44}$$

and

$$\frac{\partial X}{\partial \mu}(\lambda_l, \mu_j, \eta_k, t) = \sum_{m=-M}^{M} \left[\sum_{n=|m|}^{M} S_n^m(\eta_k, t) \frac{dP_n^m(\mu_j)}{d\mu} \right] e^{im\lambda_l}. \tag{45}$$

It is also necessary to have transform expressions relating the wind field to vorticity and divergence. The scaled wind components U and V are defined in terms of the stream function ψ and velocity potential χ as

$$U = \frac{1}{a^2}\frac{\partial \chi}{\partial \lambda} - \frac{1-\mu^2}{a^2}\frac{\partial \psi}{\partial \mu}, \tag{46}$$

and

$$V = \frac{1-\mu^2}{a^2}\frac{\partial \chi}{\partial \mu} + \frac{1}{a^2}\frac{\partial \psi}{\partial \lambda}. \tag{47}$$

ζ and D are related to ψ and χ by $\zeta = \nabla^2\psi$ and $D = \nabla^2\chi$. Spherical harmonics are eigenfunctions of the Laplacian operator ∇^2, with eigenvalues $-n(n+1)/a^2$. Therefore

$$\zeta_n^m = -\left[\frac{n(n+1)}{a^2}\right]\psi_n^m, \tag{48}$$

$$D_n^m = -\left[\frac{n(n+1)}{a^2}\right]\chi_n^m. \tag{49}$$

Combining (46) and (47) with (48) and (49) gives the spectral expansions

$$\begin{aligned} U(\lambda_l, \mu_j, \eta_k, t) &= \sum_{m=-M}^{M} -im \left[\sum_{n=|m|}^{M} \frac{1}{n(n+1)} D_n^m(\eta_k, t) P_n^m(\mu_j) \right] e^{im\lambda_l} \\ &+ (1-\mu^2) \sum_{m=-M}^{M} \left[\sum_{n=|m|}^{M} \frac{1}{n(n+1)} \zeta_n^m(\eta_k, t) \frac{dP_n^m(\mu_j)}{d\mu} \right] e^{im\lambda_l}, \end{aligned} \tag{50}$$

and

$$
\begin{aligned}
V(\lambda_l, \mu_j, \eta_k, t) &= \sum_{m=-M}^{M} -im \left[\sum_{n=|m|}^{M} \frac{1}{n(n+1)} \zeta_n^m(\eta_k, t) P_n^m(\mu_j) \right] e^{im\lambda_l} \\
&- (1-\mu^2) \sum_{m=-M}^{M} \left[\sum_{n=|m|}^{M} \frac{1}{n(n+1)} D_n^m(\eta_k, t) \frac{dP_n^m(\mu_j)}{d\mu} \right] e^{im\lambda_l}. \qquad (51)
\end{aligned}
$$

The spectral coefficients of the vorticity and the divergence can be obtained from the grid point fields of U and V by using the spectral representations, which are given by (50) and (51), the orthogonality of the expansion functions, and the zero boundary conditions of U and V at the poles. Following the procedure outlined in (42) and (43) this yields

$$
\begin{aligned}
\zeta_n^m(\eta_k, t) &= \sum_{j=1}^{(3M+1)/2} w_j \left\{ im\mathcal{F}^m \left[V(\mu_j, \eta_k, t) \right] \frac{P_n^m(\mu_j)}{1-\mu_j^2} \right. \\
&+ \left. \mathcal{F}^m \left[U(\mu_j, \eta_k, t) \right] \frac{dP_n^m(\mu_j)}{d\mu} \right\}, \qquad (52)
\end{aligned}
$$

and

$$
\begin{aligned}
D_n^m(\eta_k, t) &= \sum_{j=1}^{(3M+1)/2} w_j \left\{ im\mathcal{F}^m \left[U(\mu_j, \eta_k, t) \right] \frac{P_n^m(\mu_j)}{1-\mu_j^2} \right. \\
&- \left. \mathcal{F}^m \left[V(\mu_j, \eta_k, t) \right] \frac{dP_n^m(\mu_j)}{d\mu} \right\}, \qquad (53)
\end{aligned}
$$

where as before the $\mathcal{F}$ represents the complex Fourier coefficients of the respective wind components in the square brackets.

2.4 *Time integration*

The NOGAPS spectral model uses leapfrog time differencing with a Robert time filter. This is particularly important for the adjoint model, which must integrate backwards in time in a way that is consistent with the forward time integration scheme. An integration is started with a forward time step:

$$
X_{t=1} = X_{t=0} + dt \left[\frac{\partial X}{\partial t} \right]_{t=0}, \qquad (54)
$$

followed by leapfrog and time filter steps,

$$X_{t=2} = X_{t=0} + 2dt \left[\frac{\partial X}{\partial t}\right]_{t=1} \tag{55}$$

$$\tilde{X}_{t=1} = X_{t=1} + \epsilon \left[X_{t=2} - 2X_{t=1} + X_{t=0}\right] \tag{56}$$

then

$$X_{t=3} = X_{t=1} + 2dt \left[\frac{\partial X}{\partial t}\right]_{t=2} \tag{57}$$

$$\tilde{X}_{t=2} = X_{t=2} + \epsilon \left[X_{t=3} - 2X_{t=2} + X_{t=1}\right] \tag{58}$$

and so on. dt is the time step and ϵ is the time filter coefficient: $0 \leq \epsilon \leq 0.25$. The $\tilde{X}(t)$ are the time filtered values that replace the existing $X(t)$. Distinguishing between the $\tilde{X}(t)$ and $X(t)$ is an important detail for the adjoint of the time integration scheme.

3 The tangent linear model

The linearization of the set of equations given in section 2 above is straightforward. The only novel feature of the resulting linear system of equations is that they describe the behavior of perturbations about a basic state which varies in time and all three space dimensions. This means that the coefficients of the linear model that depend on this basic state will be tangent to the nonlinear model trajectory in phase space, hence the term tangent linear model, or TLM. The TLM is nearly always an extremely close approximation to the nonlinear model unless the linear approximation breaks down even for very short time and space scales.

A very important practical implication of the time varying basic state is that during a trajectory generating non-linear model integration the forecast variables (coefficients) must be writing to a file at sufficiently high frequency to resolve the trajectory. This is often every time step of the non-linear model integration. Then during a TLM model (or adjoint model) integration based on this trajectory the file must be read at this frequency to retrieve the

coefficients in the appropriate order. This implies a tremendous I/O bandwidth overhead for TLM and adjoint model integrations far in excess of normal NWP model integrations. During singular vector calculations this I/O requirement can be extremely burdensome and often requires alternate strategies such as storing the trajectory in memory, if possible, or using special I/O devices with much better performance than typical rotating mass storage devices. This extra burden on TLM and adjoint integrations led Errico *et al.* (1993) to discuss the impact of less frequent trajectory update on TLM integrations.

To demonstrate a simple linearization example, consider the simple quadratically nonlinear equation $y = ab$. Assume a, b, and y are composed of basic state components $\overline{a}$, $\overline{b}$, $\overline{y}$ and perturbations a', b', and y' about the basic state. We have

$$\overline{y} + y' = (\overline{a} + a')(\overline{b} + b').$$

Collecting terms and assuming $\overline{y} = \overline{a}\overline{b}$ yields

$$y' = \overline{a}b' + \overline{b}a' + a'b'.$$

The linearization is completed by neglecting the $a'b'$ term, which should be small compared to the linear terms if perturbations are small. For higher order nonlinearities there would be additional terms to be neglected, but the assumption of small perturbations relative to the basic state variables is still the basis of successful linearization.

We shall derive tangent linear expansions of each of the important governing equations presented in section 2(b) above. Intermediate variables will be defined where convenient and to assist clarity. An important issue in TLM and adjoint model development is coding order, since the adjoint model is a series of matrix transposes which are equivalent to exact reversal of the logical flow of expressions in the TLM model. We will therefore present equations in essentially the same order they occur in the TLM model code. The complete TLM code is available upon request from the author. We will follow the normal convention of representing a perturbation variable as $(\ )'$ and a basic state variable as $\overline{(\ )}$. A nonlinear

NOGAPS model variable expansion in discrete form is therefore

$$X_{ljk}(t) = \overline{X}_{ljk}(t) + X'_{ljk}(t),$$

where l, j, k are the indices for longitude, latitude, and vertical coordinate, respectively. Note the full space and time dependence of the basic state. For convenience in the following we will drop all but the vertical index.

3.1 *Spectral to real transforms*

The time stepping process in the NOGAPS spectral model is done with the variables defined in 'spectral space'. The same is true for the NOGAPS TLM and adjoint models. Therefore at the beginning of each TLM time step spectral variables must be transformed from spectral to 'grid point space' to allow calculation of linearized grid point tendencies. To get grid point π, ζ, D, θ, and q we use (40). Horizontal gradients of dependent variables are found using (44) and (45). For the vector wind components U and V (50) and (51) are used. Since these are linear transforms the TLM model needs no changes from the nonlinear NOGAPS code. The basic state variables are also carried in spectral form so equivalent transforms must be made to get grid point basic state quantities. With these three-dimensional grid point fields for the dependent variables we can compute all necessary diagnostic quantities and TLM tendencies.

3.2 *Terrain pressure dependent linearizations*

The perturbation forms of (21) and (22) are

$$p'_{k+1/2} = B_{k+1/2}\pi', \tag{59}$$

and

$$\Delta p'_k = \Delta B_k \pi'. \tag{60}$$

Note that for linear equations such as these the basic state does not appear. The surface pressure tendency (23) yields

$$\begin{aligned} \frac{\partial \pi'}{\partial t} &= -\sum_{k=1}^{L} \left[\Delta A_k D'_k + \Delta B_k \left(\frac{\overline{U}_k}{1-\mu^2} \frac{\partial \pi'}{\partial \lambda} + \overline{V}_k \frac{\partial \pi'}{\partial \mu} + \overline{D}_k \pi' \right.\right. \\ &+ \left.\left. \frac{1}{1-\mu^2} \frac{\partial \overline{\pi}}{\partial \lambda} U'_k + \frac{\partial \overline{\pi}}{\partial \mu} V'_k + \overline{\pi} D'_k \right) \right], \end{aligned} \tag{61}$$

and (24)

$$\begin{aligned} \left[\dot{\eta} \frac{\partial p}{\partial \eta} \right]'_{k+1/2} &= -\sum_{k=1}^{k} \left[\Delta A_k D'_k + \Delta B_k \left(\frac{\overline{U}_k}{1-\mu^2} \frac{\partial \pi'}{\partial \lambda} + \overline{V}_k \frac{\partial \pi'}{\partial \mu} + \overline{D}_k \pi' \right.\right. \\ &+ \left.\left. \frac{1}{1-\mu^2} \frac{\partial \overline{\pi}}{\partial \lambda} U'_k + \frac{\partial \overline{\pi}}{\partial \mu} V'_k + \overline{\pi} D'_k \right) \right] - B_{k+1/2} \frac{\partial \pi'}{\partial t}. \end{aligned} \tag{62}$$

Note the general characteristic of the linearized equations to be more complex than their nonlinear counterparts, typically with each nonlinear term becoming two or more linear terms. Therefore the TLM is significantly more expensive to run that the dry version of the nonlinear NOGAPS.

Linearization of the transcendental Exner functions (25) and (26) is most convenient with a first order Taylor series expansion. Because P is only a function of the terrain pressure π, we have

$$P' = \overline{\left(\frac{\partial P}{\partial \pi} \right)} \pi',$$

so the half level perturbation Exner function is

$$P'_{k+1/2} = \frac{\kappa B_{k+1/2} \overline{P}_{k+1/2}}{\overline{p}_{k+1/2}} \pi'. \tag{63}$$

At full levels using (27) yields

$$P'_k = \left(\frac{B_{k+1/2} \left(\overline{P}_{k+1/2} - \overline{P}_k \right) + B_{k-1/2} \left(\overline{P}_k - \overline{P}_{k-1/2} \right)}{\Delta \overline{p}_k} \right) \pi'. \tag{64}$$

To linearize (27) we use (60), (63), and (64):

$$\begin{aligned}\left[\frac{\partial P_k}{\partial \pi}\right]' &= \left[\frac{B_{k+1/2}\left(P'_{k+1/2}-P'_k\right)+B_{k-1/2}\left(P'_k-P'_{k-1/2}\right)}{\Delta\overline{p}_k}\right] \\ &- \left[\frac{B_{k+1/2}\left(\overline{P}_{k+1/2}-\overline{P}_k\right)+B_{k-1/2}\left(\overline{P}_k-\overline{P}_{k-1/2}\right)}{\Delta\overline{p}_k^2}\right]\Delta B_k\pi'. \qquad (65)\end{aligned}$$

3.3 *The hydrostatic equation*

The linearization of (30) is

$$\mathcal{W}'_{k+1/2}=\overline{\mathcal{W}}_{k+1/2}\left[\frac{P'_{k+1/2}-P'_k}{\overline{P}_{k+1/2}-\overline{P}_k}-\frac{P'_{k+1}-P'_k}{\overline{P}_{k+1}-\overline{P}_k}\right], \qquad (66)$$

which combined with (29) leads to

$$\theta'_{k+1/2}=\theta'_{k+1}+\mathcal{W}'_{k+1/2}(\overline{\theta}_k-\overline{\theta}_{k+1})+\overline{\mathcal{W}}_{k+1/2}\left(\theta'_k-\theta'_{k+1}\right). \qquad (67)$$

The hydrostatic equation (39) is linearized using the definition for $\theta'_{k+1/2}$, $P'_{k+1/2}$, and P'_k given in (67), (63), and (64). We have

$$\left.\begin{aligned}\phi'_k &= c_p\theta'_k(\overline{P}_{k+1/2}-\overline{P}_k)+c_p\overline{\theta}_k(P'_{k+1/2}-P'_k), && k=L \\ \phi'_k-\phi'_{k+1} &= c_p\theta'_{k+1/2}(\overline{P}_{k+1}-\overline{P}_k)+c_p\overline{\theta}_{k+1/2}(P'_{k+1}-P'_k), && L-1\geq k\geq 1\end{aligned}\right\}. \qquad (68)$$

The discrete hydrostatic equation is an integral of atmospheric mass from the Earth's surface upward. (68) shows the dependence of a layer k on layers below it, e.g. $k+1$. The ordering of the k index is also explicitly shown. Reversing this order will be critical for the adjoint of the TLM hydrostatic equation.

3.4 *The potential temperature and moisture equations*

To linearize any of the multi-level prognostic equations such as (31) we need a linearization of the vertical advection operator (28). Using (60) and (62) the perturbation form of (28) is

$$\left(\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial X}{\partial p}\right)'_k=\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]'_{k+1/2}\left(\frac{\overline{X}_{k+1/2}-\overline{X}_k}{\Delta\overline{p}_k}\right)+\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]'_{k-1/2}\left(\frac{\overline{X}_k-\overline{X}_{k-1/2}}{\Delta\overline{p}_k}\right)$$

$$
\begin{aligned}
&- \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2}\left(\frac{\overline{X}_{k+1/2}-\overline{X}_k}{(\Delta\overline{p}_k)^2}\right)\pi' - \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}\left(\frac{\overline{X}_k-\overline{X}_{k-1/2}}{(\Delta\overline{p}_k)^2}\right)\pi' \\
&+ \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2}\left(\frac{X'_{k+1/2}-X'_k}{\Delta\overline{p}_k}\right) + \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}\left(\frac{X'_k-X'_{k-1/2}}{\Delta\overline{p}_k}\right). \qquad (69)
\end{aligned}
$$

With (69) the perturbation forms of (31) and (38) are

$$
\frac{\partial \theta'_k}{\partial t} = \overline{D}_k\theta'_k + \overline{\theta}_k D'_k - \alpha\left(U'_k\overline{\theta}_k, V'_k\overline{\theta}_k\right) - \alpha\left(\overline{U}_k\theta'_k, \overline{V}_k\theta'_k\right) - \left(\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial \theta}{\partial p}\right)'_k + Q'_{\theta_k}, \qquad (70)
$$

and

$$
\frac{\partial q'_k}{\partial t} = \overline{D}_k q'_k + \overline{q}_k D'_k - \alpha\left(U'_k\overline{q}_k, V'_k\overline{q}_k\right) - \alpha\left(\overline{U}_k q'_k, \overline{V}_k q'_k\right) - \left(\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial q}{\partial p}\right)'_k + Q'_{q_k}, \qquad (71)
$$

Because we only has a dry TLM at this point, the Q'_{θ_k} and Q'_{q_k} terms only contain a simple vertical mixing process in the planetary boundary layer (PBL). No perturbation surface sensible or latent heat fluxes are included. Future applications will require the inclusion of these fluxes as well as some linearized moist physics parameterization. The simplified linear mixing parameterization is described in section (f) below.

3.5 *The vorticity and divergence equations*

The perturbation forms of the ζ and D tendency equations (33) and (34) are derived by first linearizing the expressions for G_k (35) ,H_k (36), and I_k (68). Using (69) and (65), we have

$$
\begin{aligned}
G'_k &= U'_k(\overline{\zeta}_k + f) + \overline{U}_k\zeta'_k + \left(\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial V}{\partial p}\right)'_k \\
&+ (1-\mu^2)\frac{c_p}{a^2}\left[\theta'_k\overline{\left(\frac{\partial P}{\partial \pi}\right)}_k\frac{\partial \overline{\pi}}{\partial \mu} + \overline{\theta}_k\left(\frac{\partial P}{\partial \pi}\right)'_k\frac{\partial \overline{\pi}}{\partial \mu} + \overline{\theta}_k\overline{\left(\frac{\partial P}{\partial \pi}\right)}_k\frac{\partial \pi'}{\partial \mu}\right] \\
&- Q'_{V_k}\frac{\cos\varphi}{a}, \qquad (72) \\
H'_k &= V'_k(\overline{\zeta_k} + f) + \overline{V}_k\zeta'_k - \left(\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]\frac{\partial U}{\partial p}\right)'_k
\end{aligned}
$$

$$
\begin{aligned}
&- \frac{c_p}{a^2}\left[\theta'_k\overline{\left(\frac{\partial P}{\partial \pi}\right)}_k\frac{\partial \overline{\pi}}{\partial \lambda} + +\overline{\theta}_k\left(\frac{\partial P}{\partial \pi}\right)'_k\frac{\partial \overline{\pi}}{\partial \lambda} + +\overline{\theta}_k\overline{\left(\frac{\partial P}{\partial \pi}\right)}_k\frac{\partial \pi'}{\partial \lambda}+\right] \\
&+ Q'_{U_k}\frac{\cos\varphi}{a}, \qquad (73)\\
I'_k &= \frac{a^2}{1-\mu^2}(U'\overline{U}_k + V'\overline{V}_k). \qquad (74)
\end{aligned}
$$

With these definitions the TLM tendency equations for ζ and D are

$$
\begin{aligned}
\frac{\partial \zeta'_k}{\partial t} &= -\alpha\left(G'_k, H'_k\right), \qquad (75)\\
\frac{\partial D'_k}{\partial t} &= \alpha\left(H'_k, -G'_k\right) - \nabla^2\left(\phi'_k + I'_k\right). \qquad (76)
\end{aligned}
$$

The diabatic forcing terms Q'_{V_k} and Q'_{U_k} in G'_k and H'_k, respectively, could potentially include perturbation momentum fluxes by model parameterizations such as gravity wave drag and cumulus friction. Currently, however, the terms only include a simplified planetary boundary layer parameterization that will give us a perturbation surface drag and vertical mixing of momentum contribution to prevent the growth of spurious unstable disturbances in the lowest model levels (Buizza 1994). The form of these forcing terms is given in section (f) below.

3.6 *Diabatic forcing terms*

The surface flux and vertical mixing parameterizations in the NOGAPS spectral model are based on Louis (1979). Details are in Hogan *et al.*(1991). In the linearization of this scheme we will neglect buoyancy effects and make other simplifying assumptions described below.

The vertical diffusion of any variable $X = \theta, q, U$, or V is:

$$
\frac{\partial X}{\partial t} = -g\frac{\partial \mathcal{F}}{\partial p},
$$

where g is the acceleration of gravity and $\mathcal{F}$ is the vertical flux of X. We choose the following TLM discrete form:

$$
\frac{\partial X'_k}{\partial t} = -g\frac{\mathcal{F}'_{k+1/2} - \mathcal{F}'_{k-1/2}}{\Delta \overline{p}_k}, \qquad (77)
$$

where we have neglected the effects of pressure perturbations, i.e., π', on the perturbation mixing. Above the surface the half level perturbation fluxes $\mathcal{F}_{k+1/2}$ are given by

$$\mathcal{F}'_{k+1/2} = l^m_{k+1/2}\,\overline{\rho}_{k+1/2}\,\overline{u}_\star\, f_{k+1/2}\left[\frac{X'_k - X'_{k+1}}{\Delta\overline{z}_{k+1/2}}\right], \qquad 1 \le k \le L-1, \tag{78}$$

where $\overline{\rho}$ is air density, $\overline{u}_\star$ is the friction velocity, $\overline{h}_0$ is the PBL depth, $\overline{z}$ is the height above the surface, and $\Delta\overline{z}$ is model layer thickness. $l^m_{k+1/2}$ is the mixing length with an asymptotic limit λ given by

$$l^m_{k+1/2} = \frac{k_v \overline{z}_{k+1/2}}{1 + k_v \overline{z}_{k+1/2}/\lambda}, \tag{79}$$

where $k_v = 0.4$ is the Von Karman constant. $f_{k+1/2}$ is the similarity function

$$f_{k+1/2} = \exp\left(-\frac{\overline{z}_{k+1/2}}{h_0}\right), \tag{80}$$

Perturbation surface fluxes are given by

$$\mathcal{F}'_S = \frac{K\,\overline{\rho}_S\,\overline{u}_\star}{\ln(\overline{z}_L/z_0)} X'_L, \tag{81}$$

where z_0 is the surface roughness length. The $\mathcal{F}'_S$ are surface momentum flux perturbations for $X = U$ and $X = V$. We are not allowing perturbation sensible and latent heat fluxes in the TLM, so the $\mathcal{F}'_S$ are zero for $X = \theta$ and $X = q$.

There are several free length scale parameters in (78), (79), (80), and (81). In the NOGAPS TLM we use the values proposed by (Buizza 1994).

$$\begin{aligned} h_0 &= 1000.0\,\mathrm{m} \quad , \\ \lambda &= 80.0\,\mathrm{m} \quad , \\ z_0 &= 0.05\,\mathrm{m} \quad \text{(land)}, \quad z_0 = 0.0005\,\mathrm{m} \quad \text{(ocean)}. \end{aligned}$$

For simplicity we use the same value of λ for both heat and momentum mixing. A logical choice for the friction velocity $u_\star$ would be a basic state $\overline{u}_\star$ generated by the nonlinear NOGAPS model and included as part of the time and space dependent trajectory which the

TLM and adjoint models need. Buizza (1994) proposed a simpler alternative that eliminates the need for any trajectory dependence. He set

$$\overline{u}_\star = 0.5 \text{ m sec}^{-1} \text{ (land)}, \quad \overline{u}_\star = 0.2 \text{ m sec}^{-1} \text{ (ocean)}.$$

We have experimented with both approaches and find little difference in TLM/adjoint results. For simplicity reasons most of our results to date have been with the Buizza values, but the trajectory method is available as an option in the TLM and adjoint codes.

To solve (77), we use a backward implicit leapfrog time differencing scheme. This leads to a simple adjustment scheme for the dependent variables, as described in Hogan *et al.*(1991). Substituting (78) and (81) into (77), rearranging terms, and dropping the primes on the X's yields

$$a_k X_{k-1} + b_k X_k + c_k X_{k+1} = \widehat{X}_k, \tag{82}$$

where $\widehat{X}_k$ is the vertical profile of X before vertical mixing, and

$$a_1 = 0, \tag{83}$$

$$a_k = -2dt\, g \left[\frac{l^m_{k-1/2}\, \overline{\rho}_{k-1/2}\, \overline{u}_\star\, f_{k-1/2}}{\Delta\overline{p}_k \Delta\overline{z}_{k-1/2}} \right], \qquad 2 \le k \le L, \tag{84}$$

$$c_k = -2dt\, g \left[\frac{l^m_{k+1/2}\, \overline{\rho}_{k+1/2}\, \overline{u}_\star\, f_{k+1/2}}{\Delta\overline{p}_k \Delta\overline{z}_{k+1/2}} \right], \qquad 1 \le k \le L-1, \tag{85}$$

$$c_L = 0, \tag{86}$$

$$b_1 = 1 - c_1 + 2dt\, g \left[\frac{l^m_{1/2}\, \overline{\rho}_{1/2}\, \overline{u}_\star\, f_{1/2}}{\Delta\overline{p}_1 \Delta\overline{z}_{1/2}} \right], \tag{87}$$

$$b_k = 1 - a_k - c_k, \qquad 2 \le k \le L-1, \tag{88}$$

$$b_L = 1 - a_L + 2dt\, g \left[\frac{K\, \overline{\rho}_S\, \overline{u}_\star}{\Delta\overline{p}_L \ln(\overline{z}_L/z_0)} \right]. \tag{89}$$

Equation (82) is a tri-diagonal system of linear equations easily solved by Gaussian elimination. The set of coefficients (83) - (89) is appropriate for the vertical fluxes of U and V. For θ and q vertical mixing the same set is used, except that currently we have no perturbation surface fluxes of heat or moisture, so $b_L = 1 - a_L$ for these TLM variables.

3.7 *Real to spectral transforms and time stepping*

The preceding TLM development yields the linear tendencies of π, ζ, D, θ, and q in 'grid point space'. These grid point tendencies are transformed to 'spectral space' using (42), (52), and (53). The spectral tendencies are adjusted for the semi-implicit corrections as in Hogan *et al.*(1991). Finally a time step is done in spectral space as in (54) to start a model integration, or as in (55) and (56) for a leapfrog and time filter step, and the TLM model variables are ready for the next time step.

4 The adjoint model

We define the adjoint of the tangent linear model operator A as A^T. We also define A as a sequence of linear operations with two levels of abstraction. At the first level it is the product of matrices that represent the time evolution nature of a model integration, i.e.,

$$A = \prod_{k=1}^{n} A_k \tag{90}$$

where each A_k is the matrix operator at time step k in a time integration of n steps. At the second level each A_k is broken down into the product of linear operators, i.e.,

$$A_k = \prod_{l=1}^{m} a_l \tag{91}$$

where the a_l are the individual fortran DO loops and lines of code of the TLM, here shown to number m. From the formal definition of the adjoint as the matrix transpose of A we have

$$A_k^T = \prod_{l=m}^{1} a_l^T, \tag{92}$$

as the adjoint of (91) and

$$A^T = \prod_{k=n}^{1} A_k^T, \tag{93}$$

as the adjoint of (90). Notice the exact reversal of matrix product order between the TLM and adjoint operations. In the following sections we will preserve this ordering in the discussion of the adjoint code development. The special case of the self-adjoint spectral transforms will be discussed separately after the rest of the adjoint model is presented.

There are two approaches to developing adjoint codes at the practical level, by manual methods and with automatic adjoint generating software. Each approach has its strengths and weaknesses, and a strategic combination of the two was used for the NOGAPS adjoint code development, as discussed below.

4.1 *Time step and time filter adjoint*

In the manual adjoint coding method each DO loop of the TLM is written as a matrix operation, the matrix is transposed, and an adjoint DO loop is written based on the transpose matrix. As an example consider the leapfrog time step and time filter of (55) and (56). In practice, of course, this is not how it would be programmed in a numerical model. In NOGAPS we have

$$X' = X^o + 2dtX', \tag{94}$$

$$X^o = X^n + \epsilon(X' - 2X^n + X^o), \tag{95}$$

$$X^n = X', \tag{96}$$

where X' is the tendency, $\partial X/\partial t$, overwritten with the 'new' value of X, X^o is the 'old' time level, and X^n is the 'now' time level. In matrix form (94) expands to

$$\begin{bmatrix} X^o \\ X' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 2dt \end{bmatrix} \begin{bmatrix} X^o \\ X' \end{bmatrix}, \tag{97}$$

(95) becomes

$$\begin{bmatrix} X' \\ X^n \\ X^o \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \epsilon & 1-2\epsilon & \epsilon \end{bmatrix} \begin{bmatrix} X' \\ X^n \\ X^o \end{bmatrix}, \tag{98}$$

and (96) becomes

$$\begin{bmatrix} X' \\ X^n \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} X' \\ X^n \end{bmatrix}. \tag{99}$$

Following the steps outlined above for adjoint code ordering, we can write

$$\begin{bmatrix} X'_a \\ X^n_a \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} X'_a \\ X^n_a \end{bmatrix}, \tag{100}$$

$$\begin{bmatrix} X'_a \\ X^n_a \\ X^o_a \end{bmatrix} = \begin{bmatrix} 1 & 0 & \epsilon \\ 0 & 1 & 1-2\epsilon \\ 0 & 0 & \epsilon \end{bmatrix} \begin{bmatrix} X'_a \\ X^n_a \\ X^o_a \end{bmatrix}, \tag{101}$$

$$\begin{bmatrix} X^o_a \\ X'_a \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 2dt \end{bmatrix} \begin{bmatrix} X^o_a \\ X'_a \end{bmatrix}, \tag{102}$$

where the X_a are adjoint variables. Expanding back to equation form yields the adjoint of (96)

$$\left.\begin{aligned} X'_a &= X'_a + X^n_a \\ X^n_a &= 0 \end{aligned}\right\}, \tag{103}$$

the adjoint of (95)

$$\left.\begin{aligned} X'_a &= X^n_a + \epsilon X^o_a \\ X^n_a &= (1-2\epsilon) X^o_a \\ X^o_a &= \epsilon X^o_a \end{aligned}\right\}, \tag{104}$$

and the adjoint of (94)

$$\left.\begin{aligned} X^o_a &= X^o_a + X'_a \\ X'_a &= 2dt X'_a \end{aligned}\right\}. \tag{105}$$

Now for some observations. Notice that in (97), (98), and (99) each left-hand side variable was assigned to the bottom row of its matrix expansion. This is required to ensure proper ordering of the adjoint expressions. Also notice that in (103), (104), and (105) the first equation is an accumulation into a variable, i.e., a variable occurs on both sides of the equal sign. Therefore care must be taken to ensure that this variable is initialized properly at the beginning of each adjoint time step. With almost no exceptions adjoint variables should be set to zero at the beginning of each time step, and often at intermediate points in the code as well. The general rule is that when in the TLM a variable is assigned a new value with no dependence on a previous value, then at the corresponding point in the adjoint code the companion adjoint variable must be set to zero. For example, in (103) the right-hand side X'_a must be set to zero for correct results from this sequence of equations. Failure to set adjoint variables to zero at appropriate times is the overwhelming source of errors in adjoint models, as this author can wearily attest.

Clearly the preceding exercise demonstrates the mechanical nature of adjoint code generation, which leads to the obvious question: what about automatic adjoint generation software? In recent years considerable effort has be devoted to such software development, and it has made adjoint model development much less arduous by relieving modelers of the tedium of hand coding the operations such as described above. For the NOGAPS adjoint model the software of Giering and Kaminski (1996) was used on nearly all of the TLM code described in part 3. Adjoint software is particularly well suited for operating on individual subroutines at the bottom of a calling tree with all communication to the rest of the model through calling arguments, i.e., no COMMON. In general the software cannot reliably produce correct adjoint code for routines which contain subroutine calls or COMMON variables. The best strategy for adjoint model development is therefore to use the automatic software for clean code areas of the TLM, and use manual methods in top level routines which contain inter-procedural communication. The fortran code produced

by the automatic software is also cumbersome and difficult to read, so hand editing is often necessary to get more esthetically pleasing code. The automatic software is also unable to recognize self-adjoint constructs such as the spectral transforms in the NOGAPS TLM, so here too manual methods must be used.

We will now proceed step by step though an adjoint model time step in the appropriate order. We will drop the subscript a for convenience, since in the subsequent sections **ALL** references to dependent variables will be to the adjoint variables. For sake of brevity not all the details will be presented. In particular the adjoints of the semi-implicit algorithms and horizontal diffusion will not be shown, since they are linear operations and are easily handled by automatic adjoint software. Each adjoint model step begins with the adjoint of the time step and time filter, which is already described above. Next is the adjoint of the grid-to-spectral transform. We shall postpone the discussion of this until the end of this section, since it is easier to understand the self-adjoint properties when both grid-to-spectral and spectral-to-grid transform adjoints are described together.

4.2 *Diabatic forcing adjoint*

The simple form for TLM perturbation surface fluxes and vertical mixing also makes the adjoint quit simple. The TLM dependent variables appear only in the tri-diagonal linear system (82). The adjoint is therefore

$$c_{k-1}X_{k-1} + b_k X_k + a_{k+1}X_{k+1} = \widehat{X}_k, \tag{106}$$

where $\widehat{X}_k$ is now the vertical profile of the adjoint variable X before the adjoint mixing is applied. The coefficients are identical to those of (83) - (89).

4.3 *Vorticity and divergence adjoint equations*

We need the adjoints of (75) and (76). In the TLM the sequence of operations is to evaluate (72), (73), and (74) in grid point space, add the diabatic forcing terms and perturbation geopotential ϕ', and then do grid-to-spectral transforms to get spectral tendencies of ζ and D. In the adjoint, therefore, we must first do the adjoint of the transform. As mentioned earlier, we shall postpone the details of this for now, except to observe that a grid-to-spectral transform adjoint is a spectral-to-grid transform with scaling. The reverse is true for a spectral-to-grid transform adjoint. For convenience we define the adjoint operators $\mathcal{R}_a$ and $\mathcal{S}_a$, where

$$R = \mathcal{R}_{\mathbf{a}} S \tag{107}$$

for the grid-to-spectral transform adjoint, and

$$S = \mathcal{S}_{\mathbf{a}} R \tag{108}$$

for the spectral-to-grid transform adjoint.

The first terms to consider are the Laplacian of energy terms in (74). In spectral space we have an adjoint expression

$$S_k = S_k + \nabla^2 \left(\partial D_k / \partial t \right), \tag{109}$$

where $\partial D_k/\partial t$ corresponds to X'_a in (105). Then we transform S_k to grid point space with the appropriate adjoint transform operator

$$R_k = \mathcal{R}_{\mathbf{a}} S_k, \tag{110}$$

and finally the adjoint is

$$U_k = U_k + \frac{a^2 \overline{U}_k}{1-\mu^2} R_k, \tag{111}$$

$$V_k = V_k + \frac{a^2 \overline{V}_k}{1-\mu^2} R_k, \tag{112}$$

$$\phi_k = \phi_k + R_k, \tag{113}$$

$$R_k = 0, \tag{114}$$

for $1 \leq k \leq L$.

Now we must find the adjoints of $\alpha\left(G'_k, H'_k\right)$ and $\alpha\left(H'_k, G'_k\right)$ in (75) and (76). The α operator is a spectral transform of grid point vector wind components to spectral vorticity and divergence. It is a generalization of the scalar transforms already discussed above and will be described in greater detail in a section below. For the sake of brevity we will proceed with the assumption that the adjoint of the α operator has transformed spectral adjoint variables $(\partial\zeta_k/\partial t)$ and $(\partial D_k/\partial t)$ to grid point G_k and H_k. Then the adjoints of (72) and (73) are, for $1 \leq k \leq L$:

vorticity

$$\zeta_k = \zeta_k + \overline{U}_k G_k + \overline{V}_k H_k, \tag{115}$$

U wind

$$U_k = U_k + \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2} - \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}}{2\Delta\overline{p}_k} H_k + (\overline{\zeta}_k + f) G_k, \tag{116}$$

$$U_{k-1} = U_{k-1} + \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}}{2\Delta\overline{p}_k} H_k, \tag{117}$$

$$U_{k+1} = U_{k+1} - \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2}}{2\Delta\overline{p}_k} H_k, \tag{118}$$

V wind

$$V_k = V_k - \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2} - \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}}{2\Delta\overline{p}_k} G_k + (\overline{\zeta}_k + f) H_k, \tag{119}$$

$$V_{k-1} = V_{k-1} - \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}}{2\Delta\overline{p}_k} G_k, \tag{120}$$

$$V_{k+1} = V_{k+1} + \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2}}{2\Delta\overline{p}_k} G_k, \tag{121}$$

vertical velocity

$$\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2} = \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2} + \frac{\overline{V}_k - \overline{V}_{k-1}}{2\Delta\overline{p}_k} G_k - \frac{\overline{U}_k - \overline{U}_{k-1}}{2\Delta\overline{p}_k} H_k, \tag{122}$$

$$\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2} = \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2} + \frac{\overline{V}_{k+1}-\overline{V}_k}{2\Delta\overline{p}_k}G_k - \frac{\overline{U}_{k+1}-\overline{U}_k}{2\Delta\overline{p}_k}H_k, \tag{123}$$

and if

$$X^a = \frac{\partial\overline{\pi}}{\partial\mu}(1-\mu^2)G_k - \frac{\partial\overline{\pi}}{\partial\lambda}H_k,$$

then the potential temperature

$$\theta_k = \theta_k + \frac{c_p}{a^2}\left[\frac{B_{k-1/2}(\overline{P}_k-\overline{P}_{k-1/2}) + B_{k+1/2}(\overline{P}_{k+1/2}-\overline{P}_k)}{\Delta\overline{p}_k}\right]X^a, \tag{124}$$

Exner functions

$$P_k = P_k + \frac{c_p\Delta B_k\overline{\theta}_k}{a^2\Delta\overline{p}_k}X^a, \tag{125}$$

$$P_{k-1/2} = P_{k-1/2} - \frac{c_p B_{k-1/2}\overline{\theta}_k}{a^2\Delta\overline{p}_k}X^a, \tag{126}$$

$$P_{k+1/2} = P_{k+1/2} + \frac{c_p B_{k+1/2}\overline{\theta}_k}{a^2\Delta\overline{p}_k}X^a, \tag{127}$$

terrain pressure

$$\begin{aligned}\pi &= \pi - \frac{\Delta B_k}{(\Delta\overline{p}_k)^2}\left[\frac{c_p}{a^2}\overline{\theta}_k\left(B_{k-1/2}(\overline{P}_k-\overline{P}_{k-1/2}) + B_{k+1/2}(\overline{P}_{k+1/2}-\overline{P}_k)\right)X^a\right.\\ &+ \left(\overline{\left[\dot{\eta}\frac{\partial p}{\partial\eta}\right]}_{k-1/2}(\overline{V}_k-\overline{V}_{k-1}) + \overline{\left[\dot{\eta}\frac{\partial p}{\partial\eta}\right]}_{k+1/2}(\overline{V}_{k+1}-\overline{V}_k)\right)0.5G_k\\ &- \left.\left(\overline{\left[\dot{\eta}\frac{\partial p}{\partial\eta}\right]}_{k-1/2}(\overline{U}_k-\overline{U}_{k-1}) + \overline{\left[\dot{\eta}\frac{\partial p}{\partial\eta}\right]}_{k+1/2}(\overline{U}_{k+1}-\overline{U}_k)\right)0.5H_k\right],\end{aligned} \tag{128}$$

terrain pressure horizontal gradients

$$\frac{\partial\pi}{\partial\mu} = \frac{\partial\pi}{\partial\mu} + \frac{c_p\overline{\theta}_k(1-\mu^2)}{a^2}\left[\frac{B_{k-1/2}(\overline{P}_k-\overline{P}_{k-1/2}) + B_{k+1/2}(\overline{P}_{k+1/2}-\overline{P}_k)}{\Delta\overline{p}_k}\right]G_k, \tag{129}$$

$$\frac{\partial\pi}{\partial\lambda} = \frac{\partial\pi}{\partial\lambda} - \frac{c_p\overline{\theta}_k}{a^2}\left[\frac{B_{k-1/2}(\overline{P}_k-\overline{P}_{k-1/2}) + B_{k+1/2}(\overline{P}_{k+1/2}-\overline{P}_k)}{\Delta\overline{p}_k}\right]H_k. \tag{130}$$

Notice how the adjoint variables are accumulated and also coupled in the vertical. They must be initialized to zero at the beginning of each adjoint time step to ensure correct results. In the actual code the U_k, V_k, θ_k, and q_k adjoints are initialized from the solution

to (106). The above equations are the result of the adjoint of the vorticity and divergence equations only, additional sensitivities will be accumulated in the adjoint equations for the other dependent variables. The actual generation of the adjoint code corresponding to these equations was by the Giering and Kaminski (1996) automatic adjoint software.

4.4 *Potential temperature adjoint equations*

In (70) we again have the α linear operator relating spectral temperature tendencies to horizontal advection. As in the previous section we assume we have the adjoint of this linear operator, so the spectral adjoint quantity $\partial\theta_k/\partial t$ yields the zonal advection adjoint g_k and the meridional advection adjoint h_k. Then we have for $1 \leq k \leq L$

$$\theta_k = \theta_k + \overline{U}_k g_k + \overline{V}_k h_k, \tag{131}$$

$$U_k = U_k + \overline{\theta}_k g_k, \tag{132}$$

$$V_k = V_k + \overline{\theta}_k h_k. \tag{133}$$

Next as in (110) we transform $\partial\theta_k/\partial t$ to grid point space.

$$X_k = \mathcal{R}_{\mathbf{a}}\left(\partial\theta_k/\partial t\right), \tag{134}$$

and the adjoint equations are:

potential temperature

$$\theta_k = \theta_k + \frac{\overline{D}_k \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2} - \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}}{\Delta\overline{p}_k} X_k, \tag{135}$$

$$\theta_{k+1/2} = \theta_{k+1/2} - \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2}}{\Delta\overline{p}_k} X_k, \tag{136}$$

$$\theta_{k-1/2} = \theta_{k-1/2} + \frac{\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}}{\Delta\overline{p}_k} X_k, \tag{137}$$

vertical velocity

$$\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2} = \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k-1/2} + \frac{\overline{\theta}_{k-1/2} - \overline{\theta}_k}{\Delta \overline{p}_k} X_k, \tag{138}$$

$$\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2} = \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2} + \frac{\overline{\theta}_k - \overline{\theta}_{k+1/2}}{\Delta \overline{p}_k} X_k, \tag{139}$$

divergence

$$D_k = D_k + \overline{\theta}_k X_k, \tag{140}$$

terrain pressure

$$\pi = \pi + \frac{\Delta B_k}{(\Delta \overline{p}_k)^2}\left(\overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k-1/2}(\overline{\theta}_{k-1/2} - \overline{\theta}_k) + \overline{\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]}_{k+1/2}(\overline{\theta}_k - \overline{\theta}_{k+1/2})\right) X_k. \tag{141}$$

We are accumulating sensitivity contributions into the adjoint variables to add to those from the vorticity and divergence adjoint equations of the previous section. Since the moisture adjoint equation for q is exactly analogous to the θ equation we will not show this set of equations. The model code for them is available upon request from the author.

4.5 *Hydrostatic equation adjoint*

The adjoint equations of the hydrostatic equation (68) are

$$\phi_{k+1} = \phi_{k+1} + \phi_k, \tag{142}$$

$$\phi_k = c_p\, \phi_k, \tag{143}$$

$$\theta_k = \theta_k + (\overline{P}_{k+1/2} - \overline{P}_k)\, \phi_k, \tag{144}$$

$$P_{k+1/2} = P_{k+1/2} + \overline{\theta}_k\, \phi_k, \tag{145}$$

$$P_k = P_k - \overline{\theta}_k\, \phi_k, \tag{146}$$

for $1 \leq k \leq L-1$ and

$$\theta_k = \theta_k + c_p(\overline{P}_{k+1/2} - \overline{P}_k)\, \phi_k, \tag{147}$$

$$P_{k+1/2} = P_{k+1/2} + c_p\overline{\theta}_k\ \phi_k, \tag{148}$$

$$P_k = P_k - c_p\overline{\theta}_k\ \phi_k, \tag{149}$$

for $k = L$.

Now the adjoints of the vertical temperature interpolation equations (66) and (67):

$$\theta_{k+1} = \theta_{k+1} + \frac{\overline{P}_{k+1} - \overline{P}_{k-1/2}}{\overline{P}_{k+1} - \overline{P}_k}\ \theta_{k+1/2}, \tag{150}$$

$$\theta_k = \theta_k + \frac{\overline{P}_{k+1/2} - \overline{P}_k}{\overline{P}_{k+1} - \overline{P}_k}\ \theta_{k+1/2}, \tag{151}$$

$$P_k = P_k - (\overline{\theta}_k - \overline{\theta}_{k+1})\frac{\overline{P}_{k+1} - \overline{P}_{k+1/2}}{(\overline{P}_{k+1} - \overline{P}_k)^2}\ \theta_{k+1/2}, \tag{152}$$

$$P_{k+1} = P_{k+1} - (\overline{\theta}_k - \overline{\theta}_{k+1})\frac{\overline{P}_{k+1/2} - \overline{P}_k}{(\overline{P}_{k+1} - \overline{P}_k)^2}\ \theta_{k+1/2}, \tag{153}$$

$$\theta_{k+1/2} = 0, \tag{154}$$

for $1 \leq k \leq L-1$.

4.6 *Vertical velocity and terrain pressure adjoints*

We begin the calculations for the adjoints of (61) and (62) as in (110) with the transform of $\partial\pi/\partial t$ to grid point space, combined with summations over the previously accumulated vertical velocity adjoint variable. We have

$$X_{k+1/2} = \sum_1^k \left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2}, \quad 1 \leq k \leq L-1, \tag{155}$$

$$X_{L+1/2} = \mathcal{R}_{\mathbf{a}}\,(\partial\pi/\partial t) - \sum_{k=1}^{L-1} B_{k+1/2}\left[\dot{\eta}\frac{\partial p}{\partial \eta}\right]_{k+1/2}. \tag{156}$$

Then the adjoint expressions are:

$$U_k = U_k - \frac{\Delta B_k}{1-\mu^2}\frac{\partial\overline{\pi}}{\partial\lambda}X_{k+1/2}, \tag{157}$$

$$V_k = V_k - \Delta B_k\frac{\partial\overline{\pi}}{\partial\mu}X_{k+1/2}, \tag{158}$$

$$
\begin{aligned}
D_k &= D_k - \Delta\overline{p}_k X_{k+1/2}, && (159)\\
\pi &= \pi - \Delta B_k \overline{D}_k X_{k+1/2}, && (160)\\
\frac{\partial\pi}{\partial\lambda} &= \frac{\partial\pi}{\partial\lambda} - \frac{\Delta B_k}{1-\mu^2}\overline{U}_k X_{k+1/2}, && (161)\\
\frac{\partial\pi}{\partial\mu} &= \frac{\partial\pi}{\partial\mu} - \Delta B_k \overline{V}_k X_{k+1/2}, && (162)\\
X_{k-1/2} &= X_{k-1/2} + X_{k+1/2}, && (163)
\end{aligned}
$$

for $L \geq k \geq 1$. Because of the recursive behavior of (163) it is essential to evaluate (158)-(163) from the bottom up, i.e. indexing k from L to 1.

4.7 *Exner function adjoints*

The Exner functions from (63) and (64) are dependent only on the terrain pressure, so the adjoint expressions are surprisingly simple. We continue accumulating terrain pressure sensitivity with:

$$
\begin{aligned}
\pi &= \pi + \frac{B_{k+1/2}(\overline{P}_{k+1/2} - \overline{P}_k) + B_{k-1/2}(\overline{P}_k - \overline{P}_{k-1/2})}{\Delta\overline{p}_k} P_k \\
&+ \frac{\kappa B_{k+1/2}\overline{P}_{k+1/2}}{\overline{p}_{k+1/2}} P_{k+1/2}, && (164)
\end{aligned}
$$

for $1 \leq k \leq L$.

At this point we have completed grid point space computations of all adjoint dependent variables. In summary, since the TLM completes a time step with the leapfrog and time filter operations, we began the process with the adjoint of these time stepping algorithms with adjoint variables in spectral space. We then transformed these to grid point space using a spectral-to-grid point adjoint transform, followed be the adjoint calculations described above in exact reverse sequence from the TLM order of calculations. The next step is to transform back to spectral space using a grid-to-spectral adjoint transform.

4.8 *Spectral transform adjoints*

The scalar grid point to spectral transform (41) can be written symbolically in matrix form as

$$\begin{bmatrix} S_1 \\ \vdots \\ S_{M^2} \end{bmatrix} = \begin{bmatrix} & & \\ & K & \\ & & \end{bmatrix} \begin{bmatrix} X_1 \\ \vdots \\ X_{mn} \end{bmatrix}, \tag{165}$$

where K is the linear operator composed of the discrete Fourier transforms and the Gaussian quadratures which converts a horizontal grid point field of mn degrees of freedom to a set of complex spherical harmonic coefficients with M^2 degrees of freedom. From the definition of the adjoint we have

$$\begin{bmatrix} \widehat{X}_1 \\ \vdots \\ \widehat{X}_{mn} \end{bmatrix} = \begin{bmatrix} & & \\ & K^T & \\ & & \end{bmatrix} \begin{bmatrix} \widehat{S}_1 \\ \vdots \\ \widehat{S}_{M^2} \end{bmatrix}, \tag{166}$$

where the $\widehat{(\ )}$ are now adjoint variables and K^T is the matrix transpose of K. After some tedious but straightforward algebra (166) can be shown to be equivalent to

$$\widehat{X}(\lambda_l, \mu_j) = w_j \sum_{m=-M}^{M} \left[\sum_{n=|m|}^{M} \widehat{S}_n^m P_n^m(\mu_j) \right] e^{im\lambda_l}. \tag{167}$$

Equation (166) is identical in form to (40) except for the Gaussian weights w_j proceeding the summations. Therefore the adjoint of the grid-to-spectral transform is the spectral-to-grid transform of the forward model with an additional scaling factor weighting each Gaussian latitude of the horizontal grid point field after the transform.

A similar examination yields the adjoint of the spectral-to-grid transform as:

$$\widehat{S}_n^m = \sum_{j=1}^{(3M+1)/2} \mathcal{F}^m \left[\widehat{X}(\mu_j) \right] P_n^m(\mu_j), \tag{168}$$

which is similar to (43) except that the Gaussian quadrature weights are absent from the summation. In the adjoint code the grid-to-spectral transform code of the forward model is

used except that an array with element values of 1.0 instead of the quadrature weights is passed as a subroutine argument.

Not surprisingly, the adjoint of the spectral vorticity and divergence to grid point wind transform is a grid-to-spectral transform analogous to (52) and (53). Deriving the adjoint expressions in matrix form as shown above, it can be shown that

$$\begin{aligned}\widehat{\zeta}_n^m &= \frac{1}{n(n+1)} \sum_{j=1}^{(3M+1)/2} \left\{ im\mathcal{F}^m \left[\widehat{V}(\mu_j)\right] P_n^m(\mu_j) \right. \\ &+ \left. (1-\mu_j^2)\mathcal{F}^m \left[\widehat{U}(\mu_j)\right] \frac{dP_n^m(\mu_j)}{d\mu} \right\}, \end{aligned} \tag{169}$$

and

$$\begin{aligned}\widehat{D}_n^m &= \frac{1}{n(n+1)} \sum_{j=1}^{(3M+1)/2} \left\{ im\mathcal{F}^m \left[\widehat{U}(\mu_j)\right] P_n^m(\mu_j) \right. \\ &- \left. (1-\mu_j^2)\mathcal{F}^m \left[\widehat{V}(\mu_j)\right] \frac{dP_n^m(\mu_j)}{d\mu} \right\}, \end{aligned} \tag{170}$$

Similarly, the adjoint of the grid point wind to spectral vorticity and divergence is a spectral-to-grid transform analogous to (50) and (51) and is

$$\begin{aligned}\widehat{U}(\lambda_l, \mu_j) &= w_j \left\{ \frac{1}{1-\mu_j^2} \sum_{m=-M}^{M} -im \left[\sum_{n=|m|}^{M} \widehat{D}_n^m P_n^m(\mu_j) \right] e^{im\lambda_l} \right. \\ &+ \left. \sum_{m=-M}^{M} \left[\sum_{n=|m|}^{M} \widehat{\zeta}_n^m \frac{dP_n^m(\mu_j)}{d\mu} \right] e^{im\lambda_l} \right\}, \end{aligned} \tag{171}$$

and

$$\begin{aligned}\widehat{V}(\lambda_l, \mu_j) &= w_j \left\{ \frac{1}{1-\mu_j^2} \sum_{m=-M}^{M} -im \left[\sum_{n=|m|}^{M} \widehat{\zeta}_n^m P_n^m(\mu_j) \right] e^{im\lambda_l} \right. \\ &- \left. \sum_{m=-M}^{M} \left[\sum_{n=|m|}^{M} \widehat{D}_n^m \frac{dP_n^m(\mu_j)}{d\mu} \right] e^{im\lambda_l} \right\}. \end{aligned} \tag{172}$$

Comparing (169) and (170) to (52), (53), and (43) suggests we scale the input $\widehat{U}$ and $\widehat{V}$ with $1-\mu_j^2$, set the quadrature weights to 1.0, call the forward model subroutine for grid point wind to spectral vorticity and divergence transforms, and then scale the output

spectral coefficients with $1.0/(n(n+1))$ to produce the correct adjoint values of $\widehat{\zeta}$ and $\widehat{D}$. Likewise for (171) and (171) we scale the input $\widehat{\zeta}$ and $\widehat{D}$ with $n(n+1)$, pass these to the forward model spectral vorticity and divergence to grid point wind transform subroutine, and then scale the output grid point winds with $w_j/(1-\mu_j^2)$ to yield the correct adjoint values $\widehat{U}$ and $\widehat{V}$. Note the symmetry of the scaling factors between the two transform directions. This is necessary if the adjoint transforms are to retain the inverse properties of the companion transforms of the forward model.

5 Testing of TLM and adjoint models

Developing tangent-linear and adjoint models from a parent non-linear NWP model such as NOGAPS requires an orderly sequence of steps to minimize the risk of errors. Formulation of the TLM model described in section 3 is straightforward but tedious. Recently automatic TLM generation software has been introduced (Giering and Kaminski, 1996), but it was not available for the NOGAPS TLM , so all coding was by hand. The testing method for NOGAPS TLM, as with any linearized model, is to examine its behavior for a range of perturbation initial conditions. The criteria used is that

$$L\epsilon \approx N(x+\epsilon) - N(x) \tag{173}$$

where L is the tangent linear operator derived from the non-linear model N and ϵ is the initial perturbation on a full initial field x. In the limit of sufficiently small ϵ, (173) will become an effective equality if the linearization is robust. As ϵ increases the non-linearity of N will cause the approximation to break down. Errico et al.(1993) shows that for an ϵ equivalent to analysis uncertainty (errors), the TLM of a mesoscale forecast model (MM4) is quite accurate for 48 and even 72 hours for winter forecast cases. We expect the NOGAPS TLM

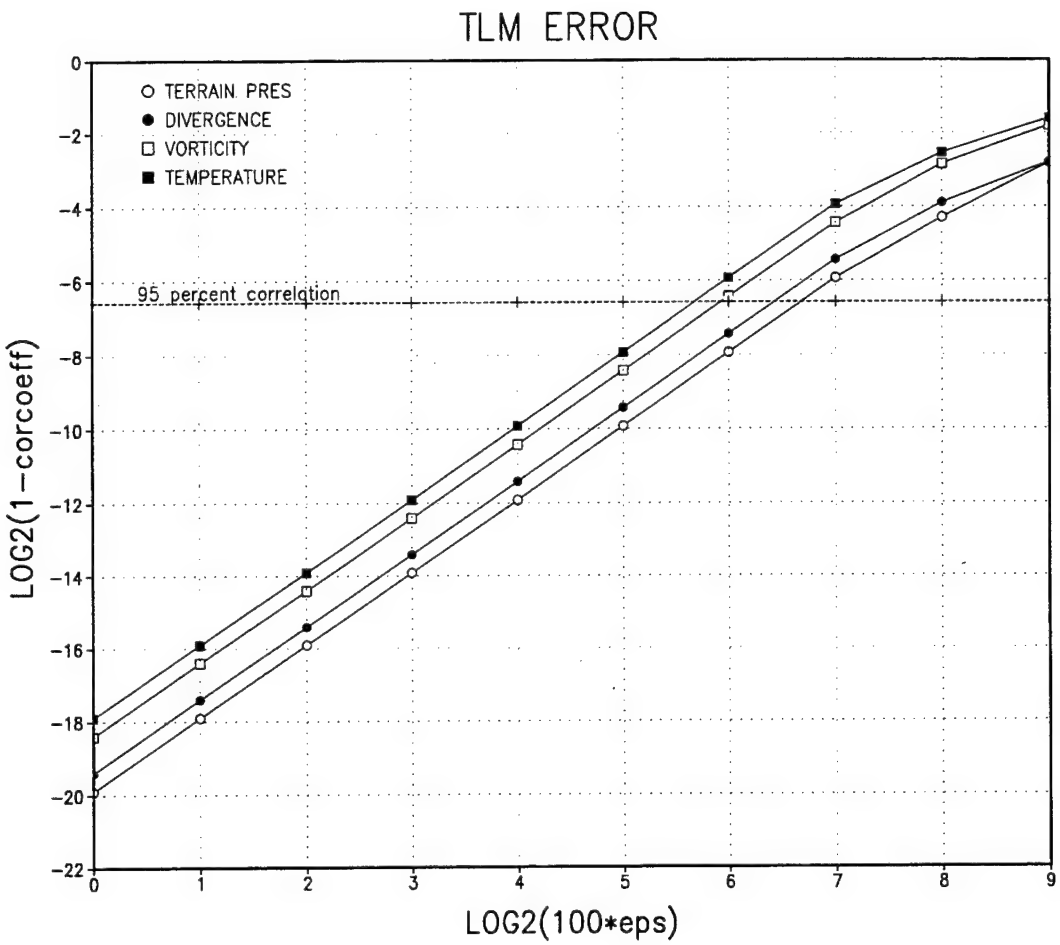

Figure 2: Correlation of TLM solution to difference between two non-linear solutions for 12 hour forecast lengths. Model resolution is T21.

to have similar capabilities. To test the NOGAPS TLM we perturb each spectral coefficient of each dependent variable randomly with values in the range $\pm\epsilon$. The correlation error between the NOGAPS TLM and difference of two non-linear (dry) forecasts as a function of ϵ is shown in Fig 2.

Fig 2 is a log-log plot of (1−correlation coefficient) vs. perturbation size. Over most of the perturbation size range the relationship is quite linear, corresponding to a quadratic growth of the correlation error as a function of ϵ. In fact, it shows that the non-linear

solutions of the dry dynamics have essentially quadratic behavior for perturbations with magnitudes nearly as large as the original unperturbed coefficients. It also shows that the TLM solution is quite accurate over this range, with better than 95% correlation over the same perturbation range. Only for extremely large perturbations does the quadratic behavior break down, probably because the cubic dependence of the vertical advection terms begins to have significant impact on the non-linear solutions. Certainly the lack of any moist physics or other discrete physical processes contributes to this surprising result, and if we repeated the experiment for a difference between two full physics non-linear forecasts we would certainly see a breakdown of the TLM solutions much sooner.

Adjoint model testing presents special problems because, unlike forward model testing, there is no simplified physical or meteorological solutions available that allow validation of some basic model properties such as dynamic balance or energy conservation. Therefore we must depend on linear algebra tests derived from the formal transpose relationship between the TLM model and its companion adjoint model. Insuring that the TLM is a correct and robust proxy for the parent nonlinear model, using the method described above, is also a top priority for developing a useful adjoint model.

The diagram below shows schematically a forward (TLM) integration from time 0 to time t, and a backward (ADJ) adjoint integration from time t to time 0. The adjoint integration is shown in terms of a gradient $\partial J/\partial X$ of a cost function J whose gradients are of interest. Typically J is based on an energy norm or some integrated measure of forecast skill such as RMS error. See Ehrendorfer and Errico (1995) for a discussion of the choice of norms.

$$\begin{array}{ccccc} X_0 & \longrightarrow & \text{TLM} & \longrightarrow & X_t \\ \frac{\partial J}{\partial X}_0 & \longleftarrow & \text{ADJ} & \longleftarrow & \frac{\partial J}{\partial X}_t \end{array}$$

In this diagram the X and $\partial J/\partial X$ represent vectors of spectral coefficients. For an

adjoint model to be the correct transpose of a TLM model, we must have

$$< X_0 \frac{\partial J}{\partial X_0} > \; = \; < X_t \frac{\partial J}{\partial X_t} > \tag{174}$$

where $<\quad>$ represents the inner product. This is called the gradient test, and is performed by making the TLM and adjoint integrations shown, saving the initial and final conditions as indicated so the inner products can be computed. For testing purposes there is no constraint on the nature of the X_0 and $\partial J/\partial X_t$ initial conditions, for the NOGAPS adjoint model random numbers were used.

The gradient test is an extremely sensitive indicator of the validity of an adjoint code. For 64 bit floating point calculations the two inner products above should have 9-10 decimal digit agreement for real data integrations of 24-48 hours. Poorer agreement than this is almost certainly due to coding errors, often because accumulated adjoint sensitivities are not initialized properly. In practice it is usually best to first do gradient testing of individual code modules for single time steps, combining modules as they are validated. When the complete adjoint model is assembled and tested for a single time step, then multiple time step testing can begin. This is the best way to isolate errors in variable initialization.

6 The singular vector system

A powerful application of TLM and adjoint models is their combination to form a singular vector matrix operator, the leading eigenvalues and eigenvectors of which describe the most unstable dynamical structures of a time-evolving basic state. This instability is manifested by the growth of some appropriately chosen norm computed from the perturbation variables of the TLM and adjoint models. The total energy norm is most commonly used. It is the sum of perturbation kinetic energy and available potential energy summed over all resolved wavenumbers and the depth of the atmosphere.

6.1 *Energy norm formulation*

At any time t the energy norm is

$$\begin{aligned} E_t &= \sum_{m=-M}^{M} \sum_{n=m}^{M} \left\{ \sum_{k=1}^{L} \left[\gamma(n,k)\, \zeta_n^m(k,t)^2 + \gamma(n,k)\, D_n^m(k,t)^2 \right.\right. \\ &+ \left.\left. \alpha(k)\, \theta_n^m(k,t)^2 \right] + \beta\, \pi_n^m(t)^2 \right\} \end{aligned} \tag{175}$$

where

$$\begin{aligned} \gamma(n,k) &= \frac{a^2 \Delta B_k}{\delta\, n(n+1)}, \\ \alpha(k) &= \frac{c_p \Delta B_k P_0(k)^2}{\delta\, T_0}, \\ \beta &= \frac{R T_0}{\delta\, \pi_0^2}, \end{aligned}$$

with $\delta = 4$ for $m = 0$ and $\delta = 2$ for $m > 0$. $P_0(k)$, T_0, and π_0 are suitable chosen reference values for Exner function, temperature, and terrain pressure, respectively.

In compact matrix form (175) is

$$E_t = x_t^T \mathbf{C}^2 x_t, \tag{176}$$

where the x is a vector of all spectral coefficient values of the perturbation variables ζ, D, θ, and π, and $\mathbf{C}^2$ is a diagonal matrix of the energy norm coefficients γ, α, and β. At time $t = 0$ the energy is

$$E_0 = x_0^T \mathbf{C}^2 x_0. \tag{177}$$

From the definitions of the TLM and adjoint model we have

$$x_t = \mathbf{L} x_0, \tag{178}$$

$$x_t^T = x_0^T \mathbf{L}^T, \tag{179}$$

which can be substituted into (176) to yield the energy at $t = t$.

$$E_t = x_0^T \mathbf{L}^T \mathbf{C}^2 \mathbf{L} x_0. \tag{180}$$

6.2 *Singular vector formulation*

We want to find the eigenvalues and eigenvectors of the linear system (180) which maximize the growth of E over the time interval t. If we normalize the initial vector x_0 such that

$$E_0 = 1, \tag{181}$$

then we can pose the algebraic eigenvalue problem in variational form as

$$J_t = x_0^T \mathbf{L}^T \mathbf{C}^2 \mathbf{L} x_0 - \lambda(x_0^T \mathbf{C}^2 x_0 - 1). \tag{182}$$

Taking the first variation of (182) yields

$$\frac{\partial J}{\partial x_0} = \mathbf{L}^T \mathbf{C}^2 \mathbf{L} x_0 - \lambda \mathbf{C}^2 x_0 = 0. \tag{183}$$

In this form (183) is awkward because it is not a symmetric eigenvalue problem. However, if we introduce an energy norm scaling transformation

$$x_0 = \mathbf{C}^{-1} z_0, \tag{184}$$

and multiply both sides by $\mathbf{C}^{-1}$, we get

$$\mathbf{C}^{-1} \mathbf{L}^T \mathbf{C}^2 \mathbf{L} \mathbf{C}^{-1} z_0 - \lambda z_0 = 0, \tag{185}$$

a much more tractable symmetric eigenvalue problem. Notice that substitution of (184) into (177) and imposing (181) yields

$$z_0^T z_0 = 1. \tag{186}$$

6.3 *The Lanczos algorithm*

The primary obstacle to solution of (185) is the enormous size of the matrix operator. As described in section 1, it can easily be of $\mathbf{O(10^7)}$ for linear operators based on models such

as NOGAPS. However, because we are interested only in a few of the leading eigenvalues of the system, we do not need an explicit representation of the operator. The Lanczos algorithm (see Golub and Van Loan (1989), Strang (1986), and Simon (1984) for details) is ideally suited for this problem, since it only requires as its input the result vector produced when a normalized input vector z_0 is passed through the matrix operator in (185). For computation purposes the matrix operator is treated as a 'black box' called by the Lanczos algorithm code as it iterates.

The Lanczos iteration process proceeds as follows, given an initial z_0 scaled to satisfy (186):

1. Compute $\mathbf{C}^{-1}z_0$. This is equivalent to scaling the z_0 back to the model dependent variables ζ, D, θ, and π.
2. Compute $\mathbf{L}\mathbf{C}^{-1}z_0$. This is the TLM integration over an appropriate forecast time, called the optimization time.
3. Compute $\mathbf{C}^2\mathbf{L}\mathbf{C}^{-1}z_0$. This is the energy norm scaling.
4. Compute $\mathbf{L}^T\mathbf{C}^2\mathbf{L}\mathbf{C}^{-1}z_0$. This is the adjoint model integration over the optimization time .
5. Compute $\mathbf{C}^{-1}\mathbf{L}^T\mathbf{C}^2\mathbf{L}\mathbf{C}^{-1}z_0$. This transforms the dependent variables back to the z_0 energy norm scaling form.

The result of step 5 is passed to the Lanczos algorithm, which re-scales the vector to yield an updated estimate for z_0, and then steps 1 - 5 are repeated. The Lanczos algorithm generates estimates for the largest eigenvalues and associated eigenvectors as these iterations proceed, both refining the estimates and increasing the number of candidate eigenvalues. Typically the number of converged eigenvalues equals about 1/3 the number of iterations. The Lanczos calculations themselves are not computationally expensive, but since each iteration involves both a TLM and adjoint model time integration, the overall computational cost is considerable. For example, A 50 iteration singular vector T47 NOGAPS calculation

for a 36 hour optimization time requires about 25000 CPU seconds on a CRAY C90 supercomputer. This includes the cost of a non-linear forecast to produce the time dependent basic state trajectory. I/O costs are also large, since the TLM and adjoint model each must read a $\sim$ 100 Mbyte trajectory file every iteration.

6.4 *The local projection operator*

As described, the singular vector system implicitly defines a global optimization domain because of the global nature of the spherical harmonic coefficients. More desirable is a regionally defined optimization domain to allow exclusion of singular vectors which lie outside an area of interest. Such a targeted domain will be influenced by a small number of singular vectors, typically 1-3, requiring only about 10 Lanczos iterations, instead of the $\sim$ 50 iterations needed to find the leading singular vectors for a global domain.

A regional optimization domain is constructed by simply discarding the variance in the model's dependent variables outside the domain, so that energy norm growth will depend only on what happens inside the domain. We define a local projection operator (LPO) as

$$\mathbf{F} = \mathcal{S}^{-1}\mathcal{R}\mathcal{S}, \tag{187}$$

where $\mathcal{S}$ is the spectral-to-grid transform operator (40), its inverse $\mathcal{S}^{-1}$ (41), and $\mathcal{R}$ is a Gaussian grid point mask equal to 1.0 in the optimization domain and 0.0 elsewhere. The mask $\mathcal{R}$ is a grid point operator, but the LPO is an operator in spherical harmonic space, so in practice there is some non-zero grid point variance due to spectral truncation Gibb's phenomena in the periphery of the projection area. This is only cosmetic, since the variance drops off very rapidly away from the projection area and does not contribute significantly to perturbation energy growth. $\mathbf{F}$ is a symmetric operator, so it can be introduced into (185), yielding the modified symmetric eigenvalue problem

$$\mathbf{C}^{-1}\mathbf{L}^T\mathbf{F}\mathbf{C}^2\mathbf{F}\mathbf{L}\mathbf{C}^{-1}z_0 - \lambda z_0 = 0, \tag{188}$$

that is still solvable with the Lanczos algorithm.

7 Example Calculations

We will demonstrate the application of the TLM, adjoint, and singular vector modeling systems on a case study from the recent Fronts & Atlantic Storm Tracks Experiment (FASTEX) (Joly *et al* 1997). The particular example includes the 'FASTEX' cyclone, an intense and rapidly moving storm that developed in the mid-Atlantic between 1997021612 UTC and 1997021912 UTC and was the subject of intense observational study during its life cycle. Figures 3a, 3b, 3c, 3d, 3e, and 3f are the NOGAPS 1997021612 UTC initial conditions and the 24, 36, 48, 60, and 72 hour forecasts respectively. The incipient FASTEX cyclone is initially visible at tau=24 as a diffuse low pressure area south of the Canadian Maritime provinces. During the next 48 hours the storm races across the Atlantic and becomes a relatively small, very intense system that brought gale force winds to Ireland, Great Britain, and the North Sea. This storm is ideally suited for study with singular vectors and adjoint sensitivities because of its relatively small size in the FASTEX validation area of the eastern Atlantic. The 72 hour 1997021612 UTC forecast shown here is the basic state trajectory chosen for the singular vector calculations.

7.1 *Singular vectors*

During FASTEX singular vector and adjoint sensitivity calculations were used to select areas for objective targeting of aircraft dropsondes in the far-upstream area from the validation zone. Because of aircraft scheduling constraints and experimental planning requirements it was necessary to run the singular vector calculations more than 24 hours before the actual targeting times. In the example shown here a 36 hour lead time was used, giving a 36 hour optimization time for the singular vectors from 1997021800 UTC to 1997021912

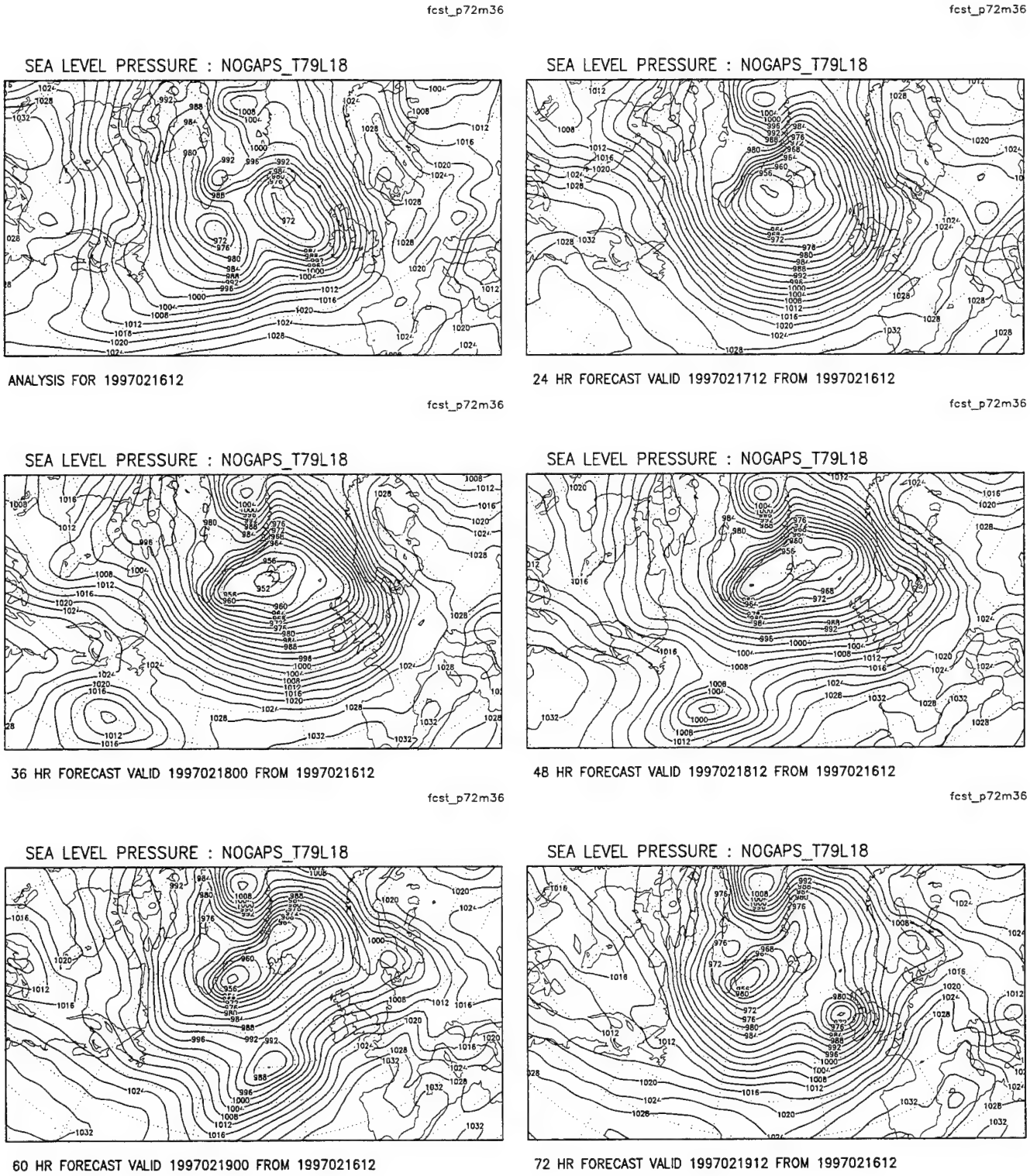

Figure 3: Time series of sea level pressure forecasts showing 72 hour development history of the 'FASTEX' cyclone.

UTC of the trajectory forecast. The 'target time' for the aircraft mission was therefore 1997021800 UTC (Tau=36 in the trajectory forecast), and the verification time 1997021912 UTC (Tau=72). Figure 4a is the perturbation temperature variable of the leading singular vector at 680 mbs for the LPO box shown bounding the FASTEX validation area. It has the largest growth of the total perturbation energy norm in the LPO. Figure 4b is this perturbation temperature after a 36 hour integration of the TLM initialized with the leading singular vector. It is the 'evolved' singular vector, and shows the growth of the perturbation energy in the LPO.

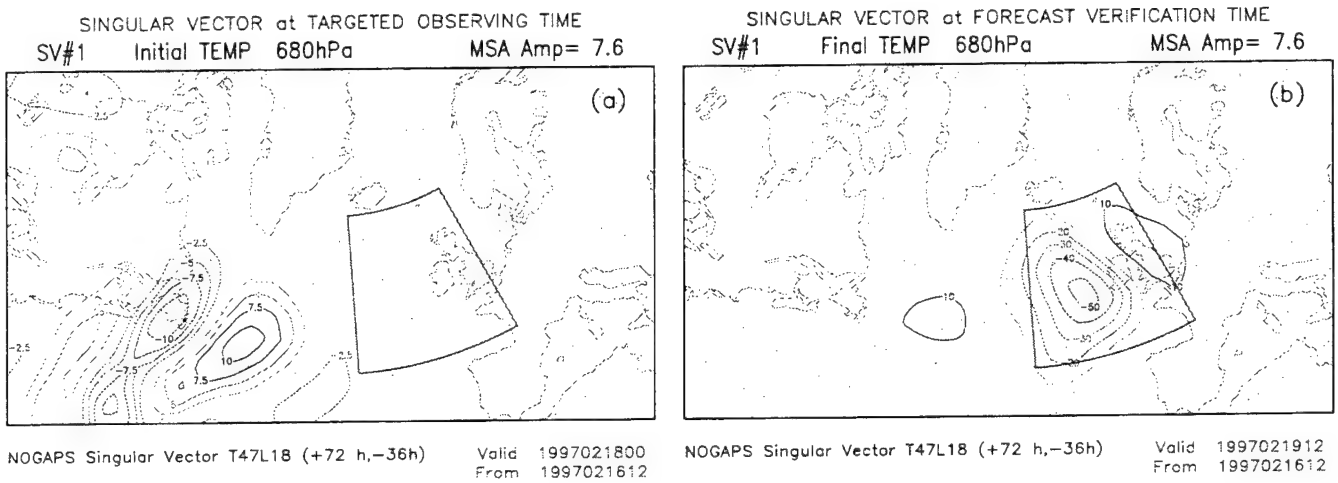

Figure 4: (a) Leading singular vector, and (b) its 36 hour TLM evolution for the FASTEX LPO. Note that the contour interval for the evolved vector is 10 times that for the initial vector.

The potential of singular vector calculations such as this is considerable. They tell us 'where the action is' in our forecast models, and by implication, in the real atmosphere. The use of an LPO allows us to focus our attention on a forecast area of interest. NWP model forecast error is the result of initial condition error growing during a forecast. This error growth is concentrated in areas of atmospheric instability such as baroclinic zones. The singular vectors tell us where these areas of instability are, and evolved vectors can

tell us where the forecast error will occur. Because the singular vectors are normalized eigenvectors, we cannot infer any magnitude or sign information about the forecast error, at least in real time, but they do tell us where improving our initial conditions, i.e., reducing the initial analysis error, will have the greatest potential for reducing forecast error. This was the premise of the objective targeting strategy in the far-upstream area of FASTEX, where aircraft dropsondes were targeted in the areas of maximum singular vector amplitude.

7.2 *Adjoint sensitivities*

Another equally powerful application of the adjoint model is computing the sensitivities of some appropriately chosen cost function J to perturbations in meteorological variables. In this example we define a cost function as the perturbation vorticity in the lower troposhere of the FASTEX LPO at a time of interest, e.g. the verification time (1997021912 UTC) of our example NWP forecast. Figure 5a is this cost function. We ask the question: what will be the sensitivity $\partial J/\partial T_0$ of this cost function to perturbations to an atmospheric variable, in this case temperature, at some earlier time, i.e., at 1997021800 UTC? The adjoint model integration is

$$\frac{\partial J}{\partial T_0} = \mathbf{L}^T \frac{\partial J}{\partial T_t}, \tag{189}$$

where the 'initial' condition $\partial J/\partial T_t$, is the derivative of our perturbation vorticity cost function at the verification time 1997021912. The sensitivity pattern shown in Figure 5b is $\partial J/\partial T_0$ at 680 mbs. Because we use the same basic state trajectory as in the singular vector calculations, and the cost function is non-zero over the same LPO used for the singular vector calculations, the resulting patterns are strikingly similar to the singular vector temperatures, although of larger horizontal scale. Adjoint sensitivity patterns such as this can be interpreted as a composite of the full spectrum of singular vectors, essentially an weighted average depending on the eigenvalues of each singular vector, which tends to

spread them out as we see. The choice of cost function is obviously an important factor in this interpretation, but that is beyond the scope of this report.

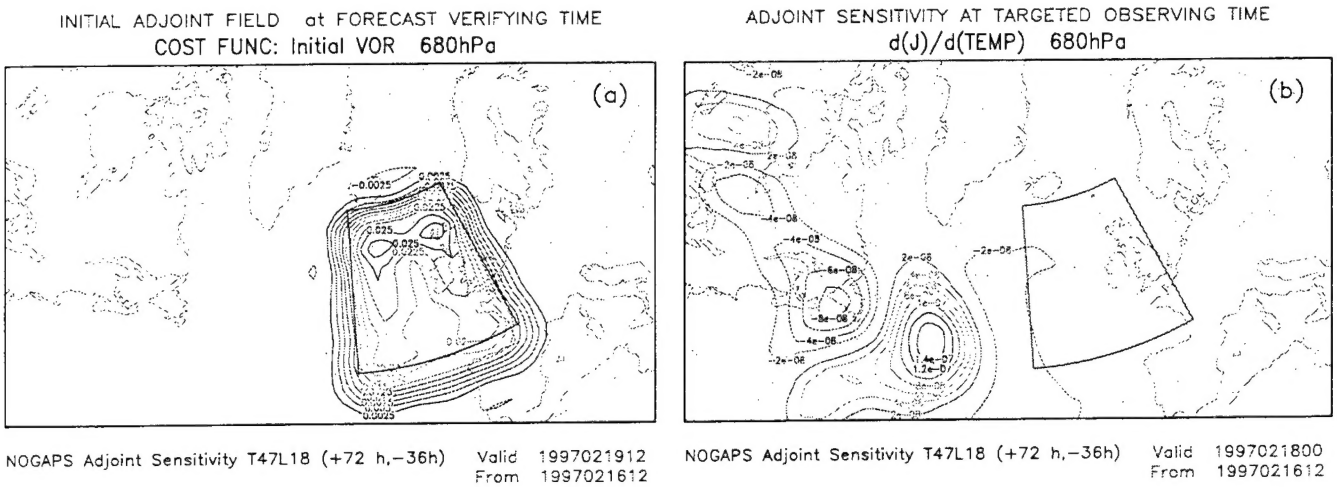

Figure 5: (a) Perturbation vorticity cost function J in FASTEX LPO at 1997021912 UTC, (b) $\partial J/\partial T$ at 1997021800 UTC.

8 Summary

The NOGAPS adjoint modeling system is a powerful new tool for understanding the behavior of the atmosphere and the NWP models used to predict it. In recent years the major operational NWP centers have reached a plateau in forecast model skill, particularly at forecast times less than 72 hours. Increasing model resolution with each new generation of computer technology, once a near guarantee of better model performance, no longer has the positive impact it once did. There are many reasons for this plateau in NWP model skill, including unrealistic parameterization of diabatic processes, model climate drift that is insensitive to resolution change, and inadequate date coverage in the tropics and southern hemisphere. Short-term forecast error (< 72 hours), however, is primarily an initial value problem, and reducing initial analysis error is the only way to improve these forecasts.

Singular vector and adjoint sensitivity calculations such as the examples in section 7 can now find the areas of collocation of analysis error and *sensitivity* to these errors. We can focus our attention on these 'hot spots' with additional observations, special attention to data quality control, or special objective analysis techniques, to reduce the analysis error.

An exhaustive discussion of other potential applications of adjoint models is far beyond the scope of this report. Errico (1997) gives very complete and detailed descriptions of the many other ways adjoints have been used to open new meteorological research areas or re-address old questions in more rigorous and efficient ways. This report should give a flavor of what real TLM and adjoint models are, and why they will be so important for the future of meteorological research.

Acknowledgments. The author owes considerable credit for the success of the NOGAPS adjoint modeling system to Ron Errico of NCAR, who provided essential tutoring on the intricacies of coding and testing TLM and adjoint models. Martin Ehrendorfer of NCAR and the University of Vienna assisted in obtaining and implementing the Lanczos algorithm software from the Numerical Algorithms Group (NAG). Rolf Langland, Ron Gelaro, and Gregg Rohaly of NRL Monterey gave many valuable suggestions and helped test the software and put it into production for FASTEX.

This work was supported by the Office of Naval Research, Program Element 0602435N.

References

[] Buizza, R., 1994: Sensitivity of optimal unstable structures. *Quart. J. Roy. Meteor. Soc.* **120**, 429-451.

[] _____, J. Tribbia, F. Molteni, and T. Palmer: 1993: Computation of optimal unstable structures for a numerical weather prediction model. *Tellus* **45A**, 388-497.

[] Charney, J. G., 1947: The dynamics of long waves in a baroclinic westerly current. *J. Meteor.* **4**, 135-162.

[] Eady, E. T., 1949: Long waves and cyclone waves. *Tellus* **1**, 33-52.

[] Ehrendorfer, M., and R. M. Errico, 1995: Mesoscale predictability and the spectrum of optimal perturbations. *J. Atmos. Sci.* **52**, 3475-3500.

[] Errico, R. M., T. Vukićević, and K. Raeder, 1993: Examination of the accuracy of a tangent linear model. *Tellus* **45A**, 462-497.

[] _____, 1997: What is an adjoint model. Submitted to *Bull. Amer. Met. Soc.*

[] Giering, R., and T. Kaminski, 1996: Recipes for adjoint code construction. Internal Report 212 from Max-Planck Institute fur Meteorologie, Hamburg, Germany. 35pp.

[] Golub, G. H., and C. F. Van Loan, 1989: *Matrix Computations.* The Johns Hopkins University Press, 642 pp.

[] Haltiner, G.J. and R.T. Williams, 1980: *Numerical Prediction and Dynamic Meteorology, Second Edition*, John Wiley and Sons, 477pp.

[] Hogan, T. F. and T. E. Rosmond: 1991: The description of the Navy Global Operational Prediction System's spectral forecast model. *Mon. Wea. Rev.* **119**, 1786-1815.

[] _____,_____, and R. Gelaro, 1991: The NOGAPS forecast model: A technical description. NOARL Report 13, Naval Research Laboratory, Stennis Space Center, MS. 220 pp.

[] Joly, A. and Collaborators, 1997: Definition of the Fronts and Atlantic Storm-Track Experiment (FASTEX), Submitted to *Bull. Amer. Meteor. Soc.*

[] Lacarra, J.-F. and O. Talagrand, 1988: Short-range evolution of small perturbations in a barotropic model. *Tellus* **40A**, 81-95.

[] Lorenz, E. N., 1965: A study of the predictability of a 28-variable atmospheric model. *Tellus* **17**, 321-333.

[] Louis, J.F., 1979: A parametric model of vertical eddy fluxes in the atmosphere. *Boundary Layer Meteorol.*, **17**, 187-202.

[] Molteni, F., and T. N. Palmer, 1993: Predictability and finite-time instability of the northern winter circulation. *Quart. J. Roy. Meteor. Soc.* **119**, 269-298.

[] Philips, N.A., 1974: Natl. Meteor Cent. Off.: Note 104. Natl. Weather Service, Washington, D.C.

[] Simmons, A. J. and R. Strüfing, 1981: An energy and angular momentum conserving finite-difference scheme, hybrid coordinates and medium-range weather prediction. ECMWF Technical Report No. 28.

[] Simon, G., 1984: The Lanczos algorithm with partial re-orthogonalization. *Math. Comput.* **42**, 115-136.

[] Strang, G., 1986: *Introduction to Applied Mathematics.* Wellesley-Cambridge Press, 758 pp.

[] Vukićević, T. and R. M. Errico, 1993: Linearization and adjoint of parameterized moist diabatic processes. *Tellus* **45A**, 493-510.